

A green intelligent routing algorithm supporting flexible QoS for many-to-many multicast

Xingwei Wang^a, Jinhong Zhang^{b,*}, Min Huang^c, Shengxiang Yang^d

^a College of Software, Northeastern University, Shenyang 110169, China

^b College of Computer Science and Engineering, Northeastern University, Shenyang 110169, China

^c College of Information Science and Engineering, Northeastern University, Shenyang 110819, China

^d Centre for Computational Intelligence (CCI), School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, United Kingdom

Abstract—The tremendous energy consumption attributed to the Information and Communication Technology (ICT) field has become a persistent concern during the last few years, attracting significant academic and industrial efforts. Networks have begun to be improved towards being "green". Considering Quality of Service (QoS) and power consumption for green Internet, a Green Intelligent flexible QoS many-to-many Multicast routing algorithm (GIQM) is presented in this paper. In the proposed algorithm, a Rendezvous Point Confirming Stage (RPCS) is first carried out to obtain a rendezvous point and the candidate Many-to-many Multicast Sharing Tree (M^2ST); then an Optimal Solution Identifying Stage (OSIS) is performed to generate a modified M^2ST rooted at the rendezvous point, and an optimal M^2ST is obtained by comparing the original M^2ST and the modified M^2ST . The network topology of Cernet2, GéANT and Internet2 were considered for the simulation of GIQM. The results from a series of experiments demonstrate the good performance and outstanding power-saving potential of the proposed GIQM with QoS satisfied.

Keywords—Power-saving, many-to-many multicast routing, dynamic niche, self-organizing learning, blind naked mole-rats food search

1. Introduction

The wide application of Information and Communication Technology (ICT), especially the ever-growing number of network access points and Internet users, drives an explosive growth in the Internet scale and an exponential rising in the network bandwidth, which causes a rapid increase on the network power consumption [1]. Various electronic devices, including network communication devices, consume a tremendous amount of power. Comparing with 530 TWh in 2005, the power consumption from ICT and consumer electronics equipment in 2012 was almost doubled, and the power requirement is estimated to rise up to about three times by 2030 [2]. Internet and its related infrastructure have contributed to more than 30% of power consumption in ICT [3]. The total world-wide electricity consumption in communication networks grew from 200 TWh in 2007 to 330 TWh in 2012 at an annual growth rate of 10.4% [4]. Such excessive power consumption of the ICT industry has led to a series of world-wide economic issues [5], and environmental concerns about collective carbon footprint from communication systems [6]. With a wide range of statistics, the Global e-Sustainability Initiative (GeSI) estimated the Green-House Gas (GHG) emissions of the ICT sector to reach about 1.43 Gtons carbon dioxide equivalent in 2020 [7].

Gupta and Singh started a seminal work on greening the Internet in 2003 [8], and suggested that the components in

network devices should be put to sleep (or into energy saving modes) when not in use in order to save energy. The current IP network resources are supplied according to the traffic demands at the peak time for the purpose to guarantee the network reliability and quite a few of network resources are redundant at most of the time, which makes the idea of green networking [9] feasible and reasonable. Recently, some effective power-efficient methods have been adopted to achieve the "greening" from two different grained perspectives: component-level and network-level power saving [10–12]. The former achieves power saving by deploying appropriate policies (e.g., Dynamic Voltage and Frequency Scaling (DVFS) in [13], Low Power Idle (LPI) in [14]) to control and adjust the working status of components in nodes or links and make their power consumption adapt to the transmitted traffic on them [15]. The latter achieves power saving from a global view of networks by improving the existing routing protocols, routing algorithms or network architectures (e.g., an enhanced version of OSPF for energy saving in [16], a power-efficient QoS routing scheme in [17], an energy-aware IP traffic engineering in [18]), or even innovating them thoroughly (e.g., a completely new design of green reconfigurable router [19]). In this paper, we consider the network-level power saving from a green routing point of view, which can relieve the energy and environment issues in the future network communication.

Meanwhile, QoS support has become more critical for current networks [20], especially with new kinds of diversified networked applications emerging in recent years, e.g., telepresence, stock quotes, PPLive, telemedicine, podcasting, twitter, and so on, which ask routing to satisfy their different QoS requirements corresponding to their specific application types [21]. Since a many-to-many multicast application, such as video conferencing, multi-players game, jam session, etc. [22], consists of multiple senders and multiple receivers with their own QoS requirements in terms of throughput, reliability, bounds on bandwidth, end-to-end delay, delay jitter, and packet loss ratio, it usually needs a strict QoS guarantee. For instance, video conferencing application has stringent reliability requirement over 99% [23], its packet loss ratio should be no more than 1%, its one-way latency should be no more than 150 ms, and its jitter should be no more than 30 ms [24]; the delay of racing game should be no more than 100 ms based on the study of two examples [25]. In addition, considering that the user QoS requirements are greatly affected by subjective factors, such as his expectation, experience, physical and mental state, personal background (age, sex, education degree),

and so on, hence are hard to be described precisely, and thus we adopt the form of interval to depict a user's inexact QoS requirements and provide flexible QoS support to him.

The multicast routing with QoS constraints is known to be NP-Complete [26], which is a relaxed problem concerned by us, i.e., the many-to-many multicast routing with power consumption minimized and user satisfaction degree on QoS maximized under QoS requirement and system capacity constraints. Thus, the problem to be solved in this paper is NP-Complete. Generally speaking, it can be solved by heuristic algorithms to generate feasible solutions within reasonable time based on some greedy functions which are nevertheless sub-optimal to the overall objective of the original problem. As a promising alternative, the intelligent optimization method can be used to find the near-optimal or optimal solution for this kind of problem by simultaneously intensifying searching in promising solution regions and exploring the uncharted space. Both the Dynamic Niche-based Self-organizing Learning Algorithm (DNSLA) [27] and Blind Naked Mole-Rats Algorithm (BNMRA) [28] are such kind of intelligent optimization method. They are introduced into our proposed algorithm, that is, the Green Intelligent flexible QoS many-to-many Multicast routing algorithm (GIQM), aiming to achieve maximum power saving under QoS constraints. GIQM is divided into two stages, that is, Rendezvous Point Confirming Stage (RPCS) and Optimal Solution Identifying Stage (OSIS). At the RPCS, GIQM obtains the rendezvous point and builds a candidate M^2ST by the self-organizing learning method, which includes three kinds of learning strategies. At the OSIS, the rendezvous point is reviewed and a modified M^2ST is obtained; then, we compared it with the candidate M^2ST obtained at the RPCS, and if it is better than the candidate, it is chosen as the solution to the problem.

To the best of our knowledge, there is no prior work that comprehensively considers both the power consumption and QoS requirements under the many-to-many multicast scenario in the backbone network with intelligent optimization methods to solve it. Our contributions in this paper are listed as follows:

- In order to make explicit power characteristics of networking elements built as metrics in routing, we introduce innovative power models of nodes and links to quantitatively describe power profiles of networking elements.
- The inexact QoS is flexibly supported, and the user satisfaction degree function on QoS is defined based on a fuzzy membership degree method.
- We formulate the aforementioned many-to-many multicast routing problem as a nonlinear programming problem, which aims to maximize user satisfaction degree on QoS and minimize network power consumption subjecting to flow and QoS constraints.
- Considering both power consumption and QoS requirements under the many-to-many multicast scenario in the backbone network, we devise a two-stage multicast routing algorithm, that is, GIQM, to solve the problem of interest.

Furthermore, our proposed GIQM focuses on routing before

the multicast is initiated, that is, the optimal multicast tree construction. If neither joining/leaving of any multicast member nor severe multicast QoS deterioration happens in the duration of the multicast, the tree keeps unchanged. When there are some multicast members joining in or leaving from the multicast, it often needs to prune the multicast tree by partial rerouting in order to add or delete the corresponding nodes from the tree with the necessary network resources allocated or released. To this, we devised the solution in [29]. If the multicast member joining or leaving happens frequently and makes the tree not optimized significantly or the multicast QoS deteriorates severely due to network congestion or other reasons, the complete rerouting can be carried out and the optimal multicast tree can be rebuilt just by our proposed GIQM.

The remainder of this paper is organized as follows. Section 2 briefly reviews the related research and background of multicast routing. Section 3 provides problem description and models power consumption of network elements. In Section 4, GIQM is described to solve the problem of interest. Section 5 presents our evaluation methodology and experimental results. Finally, conclusions are drawn in Section 6.

2. Related work

There have already been a significant amount of researches focusing on the design and implementation of routing algorithms in the Internet. Some power-saving schemes are devised toward green networking; some QoS support strategies are proposed to meet the user's requirements; furthermore, others are also developed to support different communication scenarios. Nevertheless, most of them seldom paid attention to both power saving and QoS support in the multicast scenario, especially under the many-to-many multicast scenario. We have carried out a survey of related researches as listed in Table 1, where "IO" represents Intelligent Optimization, namely Meta-Heuristics (M-H), "O2M" represents One-to-Many, "M2M" represents Many-to-Many, "●" represents that the current literature belongs to or completely supports the corresponding item, "○" represents that the current literature does not belong to or does not support the corresponding item, "◎" represents that the current literature partly supports the corresponding item. To our best knowledge, there are no specific researches on Many-to-One (M2O) multicast scenarios by far, thus there is no "M2O" column in Table 1.

A large number of works were concerned about QoS guarantees in their routing algorithms under different communication scenarios. A Multi-constrained QoS Unicast Routing Using Genetic Algorithm (MURUGA) was proposed in [30], finding the best path that satisfies multiple QoS constraints. The authors also considered the algorithm's performance in terms of search efficiency and path computation time. The authors of [31–39] instead paid their attention to QoS routing algorithms under one-to-many multicast scenario. In [31], a novel Bi-Velocity Discrete Particle Swarm Optimization (BVDPSO) algorithm was proposed to optimize one-to-many

multicast routing problem in communication networks. A Niche Ant Colony Optimization with colony guides (NACOG) algorithm was developed to tackle the minimum-cost multicast tree problem under the delay-and-bandwidth constraint in [32]. The evolutionary optimization of the NACOG algorithm was empowered with search balance between diversification and intensification. Based on the Multi-Robot Navigation Algorithm (MRNA), a swarm intelligence based QoS routing protocol was proposed for self-organizing network in [33]. It can search a routing path which satisfies the user QoS requirements and achieve the Pareto optimal utilities of the user and the network service provider under the Nash equilibrium. In [34], a Multicast Routing Algorithm based on Searching a Directed graph (MRASDH) was proposed, and an ant algorithm is applied to construct a directed sub-graph for each destination node based on the thought of deleting the non-relative nodes to shrink the searching space, and finally used a simulated annealing algorithm to obtain the qualified multicast tree in the new space. In [35], a Tree-Growth Based Ant Colony Algorithm (TGBACA) for the multi-constrained QoS multicast routing problem was proposed. It applied an ant colony optimization (ACO) algorithm to control the tree growth in order to generate a multicast tree. A Weighted Parameter for Multicast Trees (WPMT) algorithm was

proposed in [36] to solve several well-known multicast problems, including the minimum cost tree problem, the minimum delay tree problem, the delay-constrained minimum cost tree problem, the cost-constrained minimum delay tree problem, and so on. A Distributed Token-Passing based algorithm (DTP) was presented for computing a real-time sub-optimal cost multicast routing tree in [37]. It can construct a sub-optimal multicast routing tree satisfying delay constraints for all members in the multicast group. In [38], a novel path-based multicast algorithm-Qualified Group (QG) algorithm was presented for interconnection networks. QG algorithm considers the multicast latency at both the network and node levels across different traffic scenarios to achieve a high degree of parallelism during the propagation of a multicast message. In [39], a GA based algorithm for solving multimedia multicast routing was proposed, which can find the low-cost multicasting tree with bandwidth and delay constraints. It speeds up the searching ability for the optimal solution, and the global convergence of solution can be achieved by heuristic crossover and mutation operation. To sum up, intelligent optimization methods were adopted to solve the multicast routing problems in [31–35], while heuristics were applied in [36–39].

Table 1

An overview of the existing routing approaches

Literature	Power saving	QoS support		Communication scenarios	Mathematical methods		
		Flexibly	Rigidly		Multicast		IO (M-H)
				Unicast	O2M	M2M	Heuristics
Leela et al. [30]	○	○	●	●	○	○	● (MURUGA)
Shen et al. [31]	○	○	⊙ (Delay and error rate)	○	●	○	○ (BVDPSO)
Yin et al. [32]	○	○	⊙ (Bandwidth and delay)	○	●	○	○ (NACOG)
Wang et al. [33]	○	●	○	○	●	○	○ (MRNA)
Sun et al. [34]	○	○	⊙ (Delay)	○	●	○	○ (MRASDH)
Wang et al. [35]	○	○	●	○	●	○	○ (TGBACA)
Kim et al. [36]	○	○	⊙ (Delay)	○	●	○	● (TM+WPMT)
Huang et al. [37]	○	○	⊙ (Delay)	○	●	○	● (DTP)
Al-Dubai et al. [38]	○	○	⊙ (Delay and jitter)	○	●	○	● (QGs)
Younes [39]	○	○	⊙ (Bandwidth and delay)	○	●	○	● (GA)
Son et al. [40]	○	○	●	○	○	●	● (DCA)
Tyan et al. [41]	○	○	⊙ (Delay and jitter)	○	○	●	● (SBS, CBS)
Giroire et al. [42]	●	○	○	●	○	○	● (HILP-GreenRE)
Li et al. [43]	●	○	○	●	○	○	● (SPEED)
Yang et al. [44]	●	○	○	●	○	○	● (HBHGRA)
Szymanski [45]	●	○	⊙ (Bandwidth and delay)	●	●	○	● (UMFMC MMFMC)

Some authors devoted themselves to QoS routing algorithms under the many-to-many multicast scenario. In [40], the many-to-many multicast QoS routing problem was formulated as a concave quadratic program and binary Integer Linear Program (ILP), and a Difference of Convex functions Algorithm (DCA) and proximal decomposition technique are proposed to solve it. The delay and delay jitter constrained many-to-many multicast tree problems were formulated in [41]. A Source-Based Scheme (SBS) is devised, which at first finds a feasible multicast tree for each source node in the multicast group and then a minimum cover of these multicast trees so that there exists at least one feasible multicast tree in the minimum cover for each source node. A necessary and sufficient condition is also derived for a core-based multicast tree to be feasible, and a Core-Based Scheme (CBS) based on the derived condition is provided. However, in all the aforementioned literature, only the rigid QoS was taken into account when routing, while the power consumption factor was ignored.

Some literature embedded power consumption metrics in their routing decisions. In [42], a new heuristic algorithm based on the ILP formulation ($H_{ILP-GreenRE}$) was introduced to solve the energy-aware routing problem. A new energy-aware routing model with the support of data redundancy elimination (RE) was proposed. It was enabled within routers and it could virtually increase capacities of network links. In [43], a novel energy-efficient routing approach called Safe and Practical Energy Efficient Detour Routing (SPEED) was proposed to maximize the number of the aggregated links and then aggregate traffic for power savings in IP networks without any changes to the traditional IP packets forwarding diagram and routing protocols. In [44], a power model that can quantify the relationship between traffic volume and power consumption was presented, and three Hop-By-Hop Green Routing Algorithms (HBHGRA) which guaranteed loop-free routing and substantially reduced energy footprint in the Internet were progressively developed. Energy conservation and path stretch were jointly considered. However, QoS guarantee was not provided to the user when routing and the multicast scenario was neglected in [42–44].

Only a few works took both power consumption and QoS requirements into account in the routing algorithms. For instance, two routing algorithms subject to routing cost and rigid QoS constraints were proposed in [45], that is, a constrained Unicast-Max-Flow-Min-Cost algorithm (UMFMC) for routing unicast traffic flows and a constrained Multicast-Max-Flow-Min-Cost algorithm (MMFMC) to maximize the throughput of a one-to-many multicast tree while simultaneously minimizing energy costs. However, the flexible QoS constraints were not considered and the many-to-many multicast scenario was ignored in [45].

3. Modeling and problem formulation

3.1. Model description

3.1.1. Network model

The network model in this paper is denoted as a connected

graph $G = (V, E)$, as shown in Fig. 1, where V represents a set of vertices (namely network nodes) and E represents a set of edges (namely network links).

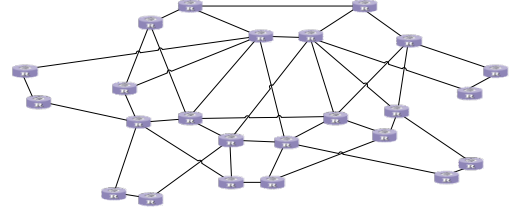


Fig. 1 Network model

3.1.2. Node model

A node model is proposed and its structure is illustrated in Fig. 2. It consists of multiple chassis, line cards, a master engine (ME), a switching fabric, multiple forwarding engines (FEs), multiple replication engines (REs), and so on. Among them, the ME is responsible for routing packets and updating routing table; the switching fabric connects input and output ports inside the router; the FE is in charge of routing table lookup; the RE is used for multicast replication. Note that all the engines are equipped with the buffer of certain size [46]. Power consumption of network node i can be formulated by Eq. (1).

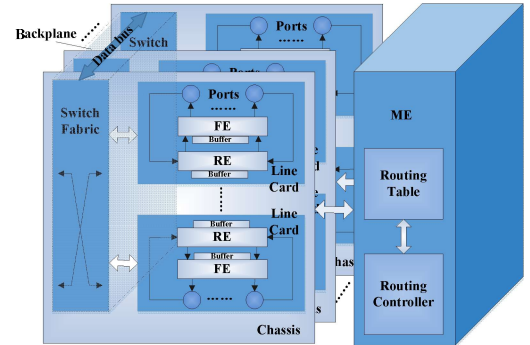


Fig. 2 The proposed node structure

$$P_{node}^i = P_{ctrl}^i + \sum_{k=1}^{N_{chass}^i} \left(P_{chass}^i \times ChaSt_k^i + \sum_{lc=1}^{N_{lc}^k} \left((P_{ford}^i + P_{repl}^i) \times LkdSt_k^{lc} + \sum_{p=1}^{N_{port}^{lc}} (P_{port}^i (1 + \alpha \times (trf_p)^{\beta}) \times PortSt_{lc}^p) \right) \right) \quad (1)$$

- P_{ctrl}^i , P_{ford}^i and P_{repl}^i represent the power consumption of the ME, a FE and a RE in node i respectively.
- P_{chass}^i and P_{port}^i represent the power consumption of a chassis and a port in node i respectively.
- N_{chass}^i represents the number of chassis in node i , N_{lc}^k represents the number of line cards in chassis k and N_{port}^{lc} represents the number of ports in line card lc .
- trf_p represents the traffic passing port p in node i .

- α and β are the constants used to denote the relationship between the traffic and power consumption.
- $ChaSt_i^k$, $LkdSt_k^{lc}$ and $PortSt_{lc}^p$ are state identifiers of components in the node, corresponding to states of chassis k , line card lc and port p in node i respectively, and their values are either 0 (sleeping state) or 1 (working state).

3.1.3. Link model

A link model is proposed and its structure is shown in Fig. 3. It has a pre-amplifier, in-line amplifiers, regenerators, a post-amplifier, and so on. Among them, the pre-amplifier is used to enhance the traffic transmitting power; an in-line amplifier is used to extend the transmission distance; a regenerator is used to provision the signal conditioning; the post-amplifier is used to convert the signal power. Power consumption of link l is calculated by the following Eq. (2).

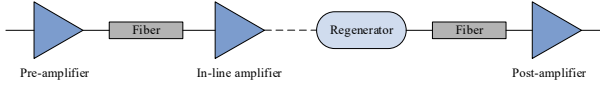


Fig. 3 The proposed link structure

$$P_{link}^l = P_{pre} + \sum_{a=1}^{N_{inl}^l} (P_{inl} \times RptrSt_l^a) + \sum_{b=1}^{N_{regen}^l} (P_{regen} \times RgenSt_l^b) + P_{post} \quad (2)$$

- P_{pre} , P_{inl} , P_{regen} and P_{post} represent the power consumption of the pre-amplifier, an in-line amplifier, a regenerator and the post-amplifier respectively.
- N_{inl}^l and N_{regen}^l represent the number of in-line amplifiers and regenerators on link l respectively.
- $RptrSt_l^a$ and $RgenSt_l^b$ denote states of in-line amplifier a and regenerator b on link l , and their values are either 0 (sleeping state) or 1 (working state).

The power consumption of a multicast tree consists of that of its nodes and links and is calculated by the following Eq. (3).

$$Power_T = \sum_{i \in T} P_{node}^i + \sum_{l \in T} P_{link}^l \quad (3)$$

3.2. QoS presentation

3.2.1. Network provided QoS

Just for simplicity, the QoS provided by the node is combined into the corresponding QoS provided by the downstream link along the route. The provided QoS is represented as a quaternion, namely $\langle \text{bandwidth (bw)}, \text{delay (dl)}, \text{delay jitter (jt)}, \text{error rate (er)} \rangle$, where bw_l , dl_l , jt_l and er_l represent the available bandwidth, delay, delay jitter and error rate of the link l respectively.

For a path P , its provided QoS is calculated in Eq. (4).

$$bw_P = \min\{bw_l \mid l \in P\}, dl_P = \sum_{l \in P} dl_l, \quad (4)$$

$$jt_P = \sum_{l \in P} jt_l, er_P = 1 - \prod_{l \in P} (1 - er_l)$$

For a tree T , its provided QoS is calculated in Eq. (5).

$$bw_T = \min\{bw_P \mid P \in T\}, dl_T = \max\{dl_P \mid P \in T\}, \quad (5)$$

$$jt_T = \max\{jt_P \mid P \in T\}, er_T = \max\{er_P \mid P \in T\}$$

3.2.2. QoS satisfaction degree

User satisfaction degree on the network provided QoS is a measure to reflect the network service quality regarding to the user QoS requirements [47]. QoS requirements of the user have been divided into K kinds of service classes according to the definition from ITU-T [48], which can be mapped into different QoS intervals. In this paper, the service classes specified in [48] are supported. We denote user QoS requirements as follows:

$$QoS_{class\ k} = (\Delta bw_{class\ k}, \Delta dl_{class\ k}, \Delta jt_{class\ k}, \Delta er_{class\ k}) \quad , \quad \text{where}$$

$$\Delta bw_{class\ k} = [bw_{class\ k}^l, bw_{class\ k}^u] \quad , \quad \Delta dl_{class\ k} = [dl_{class\ k}^l, dl_{class\ k}^u] \quad ,$$

$$\Delta jt_{class\ k} = [jt_{class\ k}^l, jt_{class\ k}^u] \quad \text{and} \quad \Delta er_{class\ k} = [er_{class\ k}^l, er_{class\ k}^u] \quad ,$$

$1 \leq k \leq K$, K is the number of the supported service classes.

For the k th class of QoS requirement, the user satisfaction degree function of bandwidth is defined by Eq. (6), which is shown in Fig. 4(a); while the user satisfaction degree function of the other three QoS parameters is defined by Eq. (7), which is shown in Fig. 4(b). In Eq. (6), $\bar{bw}_{class\ k}$ is the arithmetic mean value of $bw_{class\ k}^l$ and $bw_{class\ k}^u$; In Eq. (7), x_T can be dl_T , jt_T or er_T respectively, $x_{class\ k}^l$ can be $dl_{class\ k}^l$, $jt_{class\ k}^l$ or $er_{class\ k}^l$ respectively, $x_{class\ k}^u$ can be $dl_{class\ k}^u$, $jt_{class\ k}^u$ or $er_{class\ k}^u$ respectively, and $\bar{x}_{class\ k}$ is the arithmetic mean value of $x_{class\ k}^l$ and $x_{class\ k}^u$. Here, Ω is a positive constant; ε is a positive decimal tending to 0; ζ is a positive decimal tending to 1; $\gamma > 1$ and $0 < \eta < 1$.

$$Satis(bw_T) = \begin{cases} -\Omega & bw_T < bw_{class\ k}^l \\ \varepsilon & bw_T = bw_{class\ k}^l \\ \left(\frac{bw_T - bw_{class\ k}^l}{bw_{class\ k}^u - bw_{class\ k}^l} \right)^\gamma & bw_{class\ k}^l < bw_T < \bar{bw}_{class\ k} \\ \left(\frac{bw_T - bw_{class\ k}^l}{bw_{class\ k}^u - bw_{class\ k}^l} \right)^\eta & \bar{bw}_{class\ k} \leq bw_T < bw_{class\ k}^u \\ \zeta & bw_T = bw_{class\ k}^u \\ 1 & bw_T > bw_{class\ k}^u \end{cases} \quad (6)$$

$$Satis(x_T) = \begin{cases} -\Omega & x_T > x_{class\ k}^u \\ \varepsilon & x_T = x_{class\ k}^u \\ \left(\frac{x_{class\ k}^u - x_T}{x_{class\ k}^u - x_{class\ k}^l} \right)^\gamma & \bar{x}_{class\ k} \leq x_T < x_{class\ k}^u \\ \left(\frac{x_{class\ k}^u - x_T}{x_{class\ k}^u - x_{class\ k}^l} \right)^\eta & x_{class\ k}^l < x_T < \bar{x}_{class\ k} \\ \zeta & x_T = x_{class\ k}^l \\ 1 & x_T < x_{class\ k}^l \end{cases} \quad (7)$$

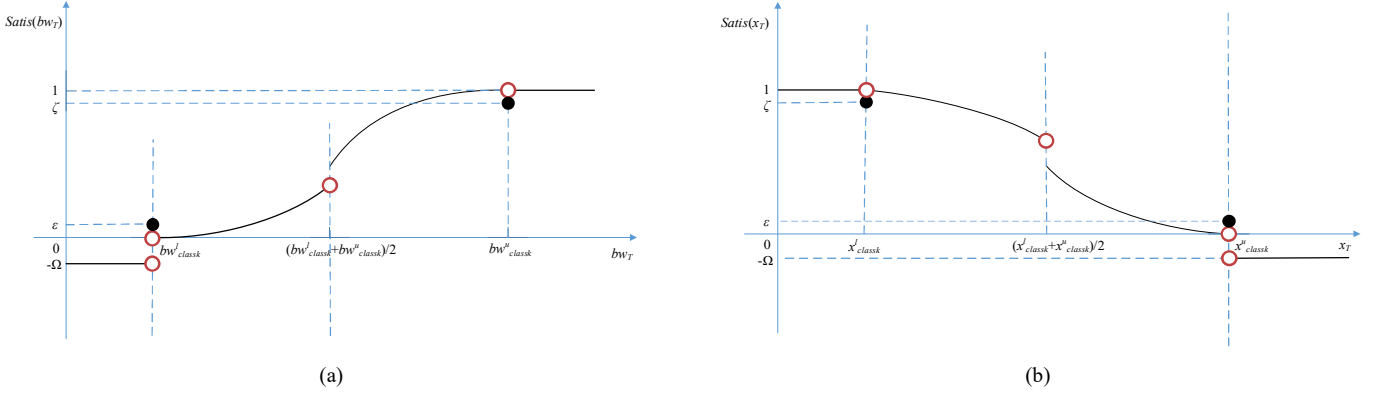


Fig. 4 The user satisfaction degree on QoS

The user satisfaction degree on the overall QoS is defined in Eq. (8), where ω_{bw} , ω_{dl} , ω_{jt} and ω_{er} are the weights of bandwidth, delay, delay jitter and error rate, representing their relative importance to the user QoS requirement respectively, and are set corresponding to different kinds of service classes, $\omega_{bw}, \omega_{dl}, \omega_{jt}, \omega_{er} \in (0,1)$ and $\omega_{bw} + \omega_{dl} + \omega_{jt} + \omega_{er} = 1$.

$$Satis(QoS_T) = \left(\omega_{bw} \times Satis(bw_T) + \omega_{dl} \times Satis(dl_T) + \omega_{jt} \times Satis(jt_T) + \omega_{er} \times Satis(er_T) \right) \quad (8)$$

3.3. Problem formulation

In this paper, the optimization objective is to maximize the user satisfaction degree on QoS and minimize the network power consumption under the user QoS requirement and system capacity constraints, as shown in Eqs. (9)–(13).

$$\text{Maximize } Satis(QoS_T) \quad (9)$$

$$\text{Minimize } Power_T \quad (10)$$

s.t.

$$\sum_{j \in T} f_{ij}^{sd} - \sum_{j \in T} f_{ji}^{sd} = \begin{cases} t^{sd}, & \forall s, d, i = s \\ -t^{sd}, & \forall s, d, i = d \\ 0, & \forall s, d, i \neq s, d \end{cases} \quad (11)$$

$$f_{ij} = \sum_{s \in T} \sum_{d \in T} f_{ij}^{sd} \leq MLU \times c_l, \forall i, j \quad (12)$$

$$bw_T \geq bw_{class}^l, dl_T \leq dl_{class}^u, jt_T \leq jt_{class}^u, er_T \leq er_{class}^u \quad (13)$$

Eqs. (11) and (12) are flow constraints under the many-to-many multicast scenario. Constraint (11) is the classical flow conservation constraint, that is, except the case of node i as a source node or a destination node, the total amount of flows going into node i is equal to the total output amount of flows going out of it. The term t^{sd} ($s, d \in T, s \neq d$) represents the average amount of traffic from the source node s to the destination node d . $f_{ij}^{sd} \in [0, t^{sd}]$ denotes the amount of flow from s to d that is routed through link l from i to j . Constraint (12) says that each link has sufficient capacity to accommodate

all the flows that traverse the link. The term c_l represents the capacity of link l . $MLU \in (0,1]$ represents the maximum link utilization that can be tolerated. The reason why we take MLU into account is as follows: on one hand, the capacity of a link should be of certain margin to accommodate the burst traffic demands especially during peak periods; on the other hand, the network should have the ability of rapidly recovering from routing failure, that is, from a practical point of view, network survivability should be considered. Hence, in this paper we investigate the performance of the proposed GIQM with $MLU=0.7$ (see details in Section 5.2). The term f_{ij} represents the total amount of flows that are routed over link l from i to j . Eq. (13) is QoS constraints.

4. Algorithm design

4.1. Introduction to DNSLA and BNMRA

DNSLA is a dynamic niche-based evolutionary algorithm, which carries out a self-organizing learning process consisting of global learning, neighborhood learning and self-learning. In the global learning, the learning rate of an individual is dynamically adjusted based on the fitness of the individual and the average fitness of the population to which the individual belongs. If the fitness of the individual is greater than the average fitness of the population, the global learning rate is enhanced to help the best individual in the current whole ecosystem to carry out an intensive search. In the neighborhood learning, the learning rate of an individual is dynamically adjusted based on the Hamming distance between the individual and the best individual of the current population to which the individual belongs. With the dynamic learning rate, the Hamming distance between them is shortened as much as possible to assist in the best individual of the current population to carry out an intensive search. In the self-learning, the learning rate of an individual is dynamically adjusted depending on the ratio of the average fitness of the population to the fitness of the individual. If the fitness of the individual is much worse than the average fitness of the population, its

self-learning rate will quickly rise to a relatively high level and obtain the dual individual by the dual mapping to the current individual. With the ongoing learning, when a niche population finds a better global solution than before, the other populations will assign some of their individuals into the existing niche population to carry out an intensive search; however, if a niche population never finds a better global solution than before during its evolutionary process, all the individuals in the population will be gradually transferred to the other niche populations and finally this niche population will disappear. Unlike a passive adaptive search strategy in traditional evolutionary algorithms, the individuals involved in DNSLA are able to learn actively. However, DNSLA is short of precise solution search ability and in some cases even falls into local optima.

BNMRA is a meta-heuristic and bionic algorithm based on social behaviors of large-scale blind naked mole-rats colony in searching the food. The suitable food sources are considered as targets to be found by employed mole-rats in the search process. When temperature and humidity change, the search of neighborhoods of food sources is carried out with different intensities. The process of collecting the food from food sources and concurrently searching their neighbors is repeated for all food sources along with their neighborhoods until no food source is available. By self-organization and division of labor (mole-rats), BNMRA has the ability of group cooperation to reach the global optimal solution with different search intensities. Furthermore, a large number of experiments conducted in [28] demonstrated that the mean and standard derivation function value of the best solution found by BNMRA was better than that of the usual optimization algorithms including GA, PSO, simulated annealing, artificial bee colony and so on, which showed that BNMRA has precise solution search ability. Consequently, BNMRA can be combined with DNSLA to obtain the optimal solution by enhancing precise solution search ability and carrying out intelligent search to avoid getting trapped in local optima.

4.2. Description of the proposed GIQM algorithm

4.2.1. Algorithm framework

On the basis of DNSLA and BNMRA, we propose the GIQM, which consists of two stages, namely RPCS and OSIS, to solve the many-to-many multicast routing problem with power consumption minimized and user satisfaction degree on QoS maximized under QoS requirement and system capacity constraints. After obtaining a set of initial solutions, the self-organizing learning method is adopted to obtain the rendezvous point and the candidate solution in RPCS. A new solution rooted at the rendezvous point, which is found in RPCS, is generated according to the selection probability depending on food production and environment information of each neighbor node in OSIS. The final solution is identified by getting the better of the two solutions found in RPCS and OSIS. The GIQM is described in the following Algorithm 1.

Algorithm 1. GIOM

Input: G, M^2R

Output: M^2ST_{OPT} or Null

- 1: Initialize parameters settings and generate initial populations;
 - 2: **Do** RPCS;
 - 3: **If** the output in RPCS is Null **then**
 - 4: **Return** Null;
 - 5: Get the candidate solution M^2ST_{RPCS} ;
 - 6: **Do** OSIS;
 - 7: **If** the output in OSIS is Null **then**
 - 8: **Return** Null;
 - 9: Get M^2ST_{OSIS} with the rendezvous point of M^2ST_{RPCS} as its root;
 - 10: Choose the better of M^2ST_{RPCS} and M^2ST_{OSIS} as the problem solution M^2ST_{OPT} ;
 - 11: **Return** M^2ST_{OPT} .
-

4.2.1.1. Solution expression

In GIQM, a solution is a many-to-many multicast tree that consists of the rendezvous point and multicast members. Assume that S and D represent the set of source nodes and destination nodes in a multicast tree respectively, namely $S = \{v_{s_i} | i = 1, 2, \dots, |S|\}$ and $D = \{v_{d_j} | j = 1, 2, \dots, |D|\}$, where $|S|$ and $|D|$ represent the number of source nodes and destination nodes in the multicast tree respectively. The solution in GIQM is expressed as a $|S| + |D|$ dimensional array, namely $x = (p_1, p_2, \dots, p_{|S|}, p'_1, p'_2, \dots, p'_{|D|})$, of which each one in the first $|S|$ elements denotes the path from the source node v_{s_i} to the rendezvous point v_{RP} namely $p_i = p(v_{s_i}, v_{RP})$, and each one in the last $|D|$ elements denotes the path from the rendezvous point v_{RP} to the destination node v_{d_j} namely $p'_j = p(v_{RP}, v_{d_j})$.

4.2.1.2. Initial solution generation

In order to obtain a set of initial solutions quickly and reasonably, at first we select a set of nodes randomly in the current network as the rendezvous points; then for each rendezvous point, we use Dijkstra algorithm to get a shortest path between the rendezvous point and each multicast member; finally, an initial solution is obtained by merging all the shortest paths with redundant branches cut and loops broken.

For example, for the rendezvous point v_{RP_0} , we find the shortest path p_{i_0} (p'_{j_0}) between v_{RP_0} and $v_{s_{i_0}}$ ($v_{d_{j_0}}$) by Dijkstra algorithm, of which $i_0 = 1, 2, \dots, |S|$ ($j_0 = 1, 2, \dots, |D|$). Then we get an initial solution as follows:

$$x_0 = (p_1, p_2, \dots, p_{i_0}, \dots, p_{|S|}, p'_1, p'_2, \dots, p'_{j_0}, \dots, p'_{|D|}).$$

4.2.1.3. Fitness function

The fitness function in GIQM is defined in Eq. (14), which is used to evaluate the quality of a solution towards the optimization objective.

$$f_T = \text{Satis}(QoS_T) / Power_T \quad (14)$$

4.2.2. RPCS

Each solution is viewed as an individual in the niche population in RPCS. Let $E = \{P_1, P_2, \dots, P_k, \dots, P_n\}$ be an ecosystem consisting of n niche populations, N_k represents the number of individuals in population P_k , P_k^i represents the i th individual in P_k . The fitness f_k^i of individual P_k^i is calculated according to Eq. (14). P_k^{best} denotes the individual with the best fitness in P_k , $\overline{f_k}$ denotes the average fitness of P_k in the current generation, $\overline{f_E}$ denotes the average fitness of the ecosystem in the current generation, P^{best} denotes the individual with the best fitness in the current ecosystem, I_{\max} denotes the largest number of iterations in RPCS.

The self-organizing learning consists of three kinds of learning strategies: global learning, neighborhood learning and self-learning, which are described below respectively.

4.2.2.1. Global learning

All the individuals in each population start to learn from the individual with the best fitness in the whole ecosystem in this phase, that is, P^{best} . The global learning rate of P_k^i is defined in Eq. (15), where $Grate$ is an initial value of the global learning rate.

$$Grate_k^i = Grate + \left(f_k^i / \overline{f_k} \right) - 1 \quad (15)$$

4.2.2.2. Neighborhood learning

All the individuals in a population P_k start to learn from the individual with the best fitness in the population in this phase, that is, P_k^{best} . The neighborhood learning rate of P_k^i is defined in Eq. (16), where $Nrate$ is an initial value of the neighborhood learning rate, $HD_{k,h}$ denotes the Hamming distance between the individual and P_k^{best} , $Length$ denotes the encoding length of identifying an individual in the ecosystem.

$$Nrate_k^i = Nrate - \left(HD_{k,h} / Length \right) + 1 \quad (16)$$

4.2.2.3. Self-learning

An individual starts to learn from itself based on dual mapping in this phase. The self-learning rate of P_k^i is defined in Eq. (17), where $Srate$ is an initial value of self-learning rate.

$$Srate_k^i = Srate \times \left(\overline{f_k} / f_k^i \right) \quad (17)$$

After I_{\max} iterations, a M^2ST called M^2ST_{RPCS} can be

obtained from the RPCS and will be reviewed in the OSIS, and v_{RP} is the rendezvous point which is the search-starting node in OSIS.

The RPCS is described in the following Algorithm 2. Here, lines 1–2 are the initialization phase; lines 5–17 are to calculate the fitness of population; lines 18–26 are the self-organizing learning process; lines 29–30 are the exchange process of the best individuals among the populations when the average fitness and the best individual of ecosystem do not change in the last generation; line 33 does pruning; lines 34–36 check constraints.

Algorithm 2. RPCS

Input: G, M^2R

Output: M^2ST_{RPCS} or Null

1: Initialize the related parameters: $n \leftarrow n_0$, $N_k \leftarrow n_k$,

$Grate \leftarrow 0.5$, $Nrate \leftarrow 0.5$, $Srate \leftarrow 0.8$, $F_k^i \leftarrow 0$, $iter \leftarrow 1$;

// n_0 and n_k denote the initial values of n and N_k respectively.

// To keep the population diversity and guarantee a reasonable evolution rate,

// we set $Grate \leftarrow 0.5$, $Nrate \leftarrow 0.5$ and $Srate \leftarrow 0.8$ according to [27].

2: Get P_k^i by Dijkstra algorithm and merging paths with redundant branches cut and loops broken;

3: **For** $iter \leftarrow 1$ to I_{\max} **do**

4: **For** $k \leftarrow 1$ to n **do**

5: $f_k^i \leftarrow \text{Satis}(QoS_k^i) / Power_k^i$; //Calculate the fitness of individuals in P_k .

6: $\overline{f_k} \leftarrow (\sum f_k^i) / N_k$; //Calculate the average fitness $\overline{f_k}$ of P_k .

7: **If** the individual with the best fitness in P_k is unique **then**

8: Set it as P_k^{best} ;

9: **Else** set the individuals with the best fitness in P_k as $P_k^{best-1}, P_k^{best-2}, \dots, P_k^{best-m_k}$;

 //Assume the number of the individuals with the best fitness in P_k is m_k .

10: Store all the individuals with the best fitness in P_k to a set of elitist solutions: $ES_k^{best} = \{P_k^{best-1}, P_k^{best-2}, \dots, P_k^{best-m_k}\}$;

11: **End-If**

12: **If** the individual with the best fitness in the ecosystem is unique **then**

13: Set it as P^{best} ;

14: **Else** set the individuals with the best fitness in the ecosystem as $P^{best-1}, P^{best-2}, \dots, P^{best-m}$;

 //Assume the number of the individuals with the best fitness in P_k is m .

15: Store all the individuals with the best fitness in the ecosystem to a set of elitist solutions:

$$ES^{best} = \{P^{best-1}, P^{best-2}, \dots, P^{best-m}\};$$

```

16:   End-If
17:    $\overline{f_E} \leftarrow (\sum \overline{f_k}) / n$  ; //Calculate the average fitness of
the ecosystem.
18:    $F_k^i \leftarrow Grate_k^i$  ; //Carry out the global learning and  $F_k^i$  is
a temporary variable to store learning rate.
19:   Update  $f_k^i$  ;
20:   If the fitness of  $P_k^i$  is not improved then
21:      $F_k^i \leftarrow Nrate_k^i$  ; //Carry out the neighborhood learning.
22:     Update  $f_k^i$  ;
23:     If the fitness of  $P_k^i$  is still not improved then
24:        $F_k^i \leftarrow Srate_k^i$  ; //Carry out the self-learning.
25:       Update  $f_k^i$  ;
26:   End-If
27: End-For
28: Update  $\overline{f_k}$ ,  $P_k^{best}$  (or  $ES_k^{best}$ ),  $\overline{f_E}$  and  $P^{best}$  (or  $ES^{best}$ );
29: If  $\overline{f_E}$  and  $P^{best}$  (or  $ES^{best}$ ) do not change before and
after this round of iteration then
30:    $P_k^{best} \leftrightarrow P_l^{best}$ ,  $l \in \{1, 2, \dots, n\}$ ,  $l \neq k$  ; //Exchange the best
individuals among populations.
31: End-For
32: Obtain  $P^{best}$  (or the element with the lowest power
consumption in  $ES^{best}$ ) with  $v_{RP}$  (s) as its(their)
rendezvous point(s);
33: Make  $P^{best}$  (or the element with the lowest power
consumption in  $ES^{best}$ ) loop-free, prune redundant
branches by deleting those useless links and thus obtain
 $M^2ST_{RPCS}$ ;
34: If constraints (11)-(13) are not satisfied by  $M^2ST_{RPCS}$  then
35:   Return Null;
36: End-If
37: Return  $M^2ST_{RPCS}$ .

```

4.2.3. OSIS

OSIS is divided into five operations: (1) calculating food production, that is, getting power consumption of a hop which includes the neighbor node v_j of the current node v_i and the link l_{ij} between them; (2) obtaining environmental information, that is, getting the network provided QoS; (3) determining search intensity; (4) calculating selection probability, and (5) searching food source(s).

The food source(s) (corresponding to the neighbor node(s)) are selected appropriately based on the following Eqs. (18)–(21). A M^2ST_{OSIS} , taking the rendezvous point v_{RP} of M^2ST_{RPCS} as its root, is gradually generated by the continuous joining of the new neighbor nodes.

Note that the mole-rats take food source(s) (neighbor node(s)) as search target. The food production of food sources is calculated by Eq. (18). T_{v_j} is a generating tree that continuously has the neighbor node v_j joined, and its power

consumption, which includes that of the rendezvous point and all the food sources, is calculated by Eq. (19).

$$P_{v_j} = P_{node}^{v_j} + P_{link}^{l_{ij}} \quad (18)$$

$$P_{T_{v_j}} = P_{node}^{v_{RP}} + \sum_{v \in T_{v_j}} P_v \quad (19)$$

The conditions of the selected food sources (corresponding to the neighbor nodes) in terms of temperature and humidity (corresponding to the network provided QoS) must always be suitable to ensure the survival of mole-rats colony (corresponding to the user QoS requirements). The user satisfaction degree on QoS of T_{v_j} is calculated according to Eq. (8). When the temperature is gradually dropping, namely the user satisfaction degree $Satis(QoS_{T_{v_j}})$ is tending to be worse than before, the search of food sources is carried out with a lower intensity. In contrast, while the temperature is close to a suitable value, namely $Satis(QoS_{T_{v_j}})$ is tending to be better than before, the search with a higher intensity will be carried out. The number A_{v_j} of food source v_j selected by v_i is defined in Eq. (20), where NEI_{v_i} denotes the set of neighbor nodes of v_i , $|NEI_{v_i}|$ denotes the number of neighbor nodes of v_i . The selection probability (SP) computed by the following Eq. (21) is set according to the obtained information (power consumption and user satisfaction degree on QoS) on food sources from the mole-rats.

$$A_{v_j} = \min \left(\left[A_{v_i} \times Satis(QoS_{T_{v_j}}) / Satis(QoS_{T_{v_i}}) \right], |NEI_{v_i}| \right) \quad (20)$$

$$SP_{v_j} = f_{T_{v_j}} / \sum_{v_j \in NEI_{v_i}} f_{T_{v_j}} \quad (21)$$

Note that the mole-rat selects a food source with high selection probability and moves towards it, that is, in the routing process, it selects the neighbor node with high selection probability as the next hop node.

The OSIS is described in the following Algorithm 3. Lines 5–12 are to calculate the selection probability SP_{v_i} ; Lines 13–15 are to select the neighbor nodes and determine the number of their next hop nodes; Lines 16–17 are to determine the number of v_j ; Lines 18–20 are to update the related parameters from the current node to the next hop node; Line 22 does pruning; Lines 23–25 check constraints.

Algorithm 3. OSIS

Input: G , M^2R , M^2ST_{RPCS}

Output: M^2ST_{OSIS} or Null

1: $T_{v_{RP}} \leftarrow \{v_{RP}\}$; //Take the rendezvous point v_{RP} of M^2ST_{RPCS} as the root of M^2ST_{OSIS} .

2: $T_{v_s} \leftarrow T_{v_{RP}}$;

3: $A_{v_{RP}} \leftarrow n_Children_{v_{RP}}$; // $n_Children_{v_{RP}}$ is the number of children nodes of v_{RP} in M^2ST_{RPCS} .

4: **While** $S_{MB} \not\subset T_{v_{RP}}$ **do** // S_{MB} is the set of multicast members.

5: **For** $v_i \in NEI_{v_{RP}}$ **do** // $NEI_{v_{RP}}$ is the set of neighbor nodes of v_{RP} .

6: **If** $T_{v_{RP}} == T_{v_i}$ **then**

7: $T_{v_i} \leftarrow T_{v_{RP}} + \{v_i\}$;

8: **End-If**

9: $P_{v_i} \leftarrow P_{node}^{v_i} + P_{link}^{v_{RP}-v_i}$;

10: $P_{v_i} \leftarrow P_{node}^{v_{RP}} + \sum_{v \in T_{v_i}} P_v$;

11: $Satis(QoS_{T_{v_i}}) \leftarrow \left(\omega_{bw} \times Satis(bw_{T_{v_i}}) + \omega_{dl} \times Satis(dl_{T_{v_i}}) \right. \\ \left. + \omega_{jt} \times Satis(jt_{T_{v_i}}) + \omega_{er} \times Satis(er_{T_{v_i}}) \right)$;

12: $Fitness_{v_i} \leftarrow (Satis(QoS_{T_{v_i}}) / P_{v_i})$;

13: $SP_{v_i} \leftarrow Fitness_{v_i} / \sum_{v_i \in NEI_{v_{RP}}} Fitness_{v_i}$; //Calculate the selection probability of v_i .

14: **End-For**

15: Select the $A_{v_{RP}}$ amount of node(s) v_i , of which selection probability or probabilities is or are the top $A_{v_{RP}}$ among those of the neighbor nodes of v_{RP} , as the next hop node(s) of v_{RP} ;

16: $Satis(QoS_{T_{v_{RP}}}) \leftarrow \left(\omega_{bw} \times Satis(bw_{T_{v_{RP}}}) + \omega_{dl} \times Satis(dl_{T_{v_{RP}}}) \right. \\ \left. + \omega_{jt} \times Satis(jt_{T_{v_{RP}}}) + \omega_{er} \times Satis(er_{T_{v_{RP}}}) \right)$;

17: $A_{v_i} \leftarrow \min \left(\left[A_{v_{RP}} \times Satis(QoS_{T_{v_i}}) / Satis(QoS_{T_{v_{RP}}}) \right], |NEI_{v_{RP}}| \right)$;

//Determine the number of the next hop node(s) of v_i .

18: $Parent_{v_i} \leftarrow v_{RP}$;

19: $T_{v_j} \leftarrow T_{v_i} + \{v_j\}$; //Add each $v_j \in NEI_{v_i}$ into multicast tree T_{v_i} respectively.

20: $T_{v_{RP}} \leftarrow T_{v_i}$, $v_{RP} \leftarrow v_i$, $NEI_{v_{RP}} \leftarrow NEI_{v_i}$, $T_{v_i} \leftarrow T_{v_j}$, $v_i \leftarrow v_j$;

//Update variables before a new round of iterations.

21: **End-While**

22: Make $T_{v_{RP}}$ loop-free, prune redundant branches by deleting those useless links and thus obtain M^2ST_{OSIS} ;

23: **If** constraints (11)–(13) are not satisfied by M^2ST_{OSIS} **then**

24: **Return Null**;

25: **End-If**

26: **Return** M^2ST_{OSIS} .

4.2.4. A simple example

Based on the topology of CERNET2 (more details are available in Section 5.1), we give an example to illustrate how to build M^2ST_{OSIS} , as shown in Fig. 5. We assume that the multicast members include SY, LZ, CS, SH and XM, and a video conferencing will be held among them. We also assume that the user QoS requirements on bandwidth, delay, delay jitter and error rate are [384Kbps, 4Mbps], [0, 400ms], [0, 200ms] and [0, 0.01] respectively, and the relative importance of bandwidth, delay, delay jitter and error rate to the user QoS are equal, that is, $\omega_{bw} = \omega_{dl} = \omega_{jt} = \omega_{er} = 0.25$. We further assume that two populations are in the ecosystem and each niche population consists of six individuals. We number all the nodes as shown in Fig. 5. We select a set of nodes randomly as the rendezvous points and then for each rendezvous point, we use Dijkstra algorithm to get the shortest path between the rendezvous point and each multicast member, and thus an initial solution is obtained by merging all the shortest paths with redundant branches cut and with loops broken. The two original populations and the corresponding fitness values of individuals are shown in Tables 2 and 3 respectively.

We calculate the user QoS satisfaction degree of each individual in the two populations by Eq. (8) and power consumption of each individual in the two populations by Eq. (3). Furthermore, we can obtain the fitness of each individual in the two populations by Eq. (14), which is shown as follows:

In the first population P_1 , $\bar{f}_1 = 0.0164$, P_1^6 is the best individual; in the second population P_2 , $\bar{f}_2 = 0.016712$, P_2^6 is the best individual; in the ecosystem E , $\bar{f}_E = 0.016556$, P_2^6 is the best individual.

Each individual except for P_2^6 in the two populations carries out a global learning and the corresponding learning rate can be calculated by Eq. (15), which is shown in Table 4. Then we obtain the new populations and their corresponding fitness values of individuals shown in Table 5 and Table 6.

Table 2

The first original population P_1 in the example

Population	Individuals	Fitness
P_1	$P_1^1 = (17-16-15-3-2, 17-17, 17-16-18-19-13-14, 17-16-18-19-13-8-9, 17-16-18-19-13-8-9-10-11)$	$f_1^1 = 0.01254$
	$P_1^2 = (10-9-8-13-3-2, 10-11-12-13-19-18-16-17, 10-11-12-13-14, 10-9, 10-11)$	$f_1^2 = 0.015576$
	$P_1^3 = (1-2, 1-2-3-15-16-17, 1-2-3-13-14, 1-2-3-13-8-9, 1-2-3-13-12-11)$	$f_1^3 = 0.016123$
	$P_1^4 = (16-15-3-2, 16-17, 16-18-19-13-14, 16-15-3-13-8-9, 16-18-19-13-12-11)$	$f_1^4 = 0.01756$
	$P_1^5 = (3-2, 3-13-19-18-16-17, 3-13-14, 3-13-12-11-10-9, 3-13-12-11)$	$f_1^5 = 0.018019$
	$P_1^6 = (5-3-2, 5-3-15-16-17, 5-3-13-14, 5-6-7-8-9, 5-6-7-8-9-10-11)$	$f_1^6 = 0.018635$

Table 3The second original population P_2 in the example

Population	Individuals	Fitness
P_2	$P_2^1=(0-1-2, 0-1-2-3-15-16-17, 0-1-2-3-13-14, 0-1-2-3-13-8-9, 0-1-2-3-13-8-9-10-11)$	$f_2^1=0.012998$
	$P_2^2=(4-2, 4-2-3-15-16-17, 4-2-3-13-14, 4-2-3-13-8-9, 4-2-3-13-12-11)$	$f_2^2=0.015734$
	$P_2^3=(19-18-16-15-3-2, 19-13-3-15-16-17, 19-13-14, 19-13-12-11-10-9, 19-13-12-11)$	$f_2^3=0.016541$
	$P_2^4=(9-8-13-3-2, 9-8-13-19-18-16-17, 9-8-13-14, 9-9, 9-10-11)$	$f_2^4=0.017695$
	$P_2^5=(6-5-3-2, 6-5-3-15-16-17, 6-7-8-13-14, 6-7-8-9, 6-7-8-9-10-11)$	$f_2^5=0.018354$
	$P_2^6=(7-6-5-3-2, 7-6-5-3-15-16-17, 7-8-13-14, 7-8-9, 7-8-9-10-11)$	$f_2^6=0.01895$

Table 4Global learning rates of all individuals except for P_2^6 in two populations

Population	Individual	Global learning rates	Population	Individual	Global learning rates
P_1	P_1^1	$Grate_1^1=0.26463415$	P_2	P_2^1	$Grate_2^1=0.27776448$
	P_1^2	$Grate_1^2=0.4497561$		P_2^2	$Grate_2^2=0.44147918$
	P_1^3	$Grate_1^3=0.48310976$		P_2^3	$Grate_2^3=0.48976783$
	P_1^4	$Grate_1^4=0.57073171$		P_2^4	$Grate_2^4=0.55882001$
	P_1^5	$Grate_1^5=0.59871951$		P_2^5	$Grate_2^5=0.59825275$
	P_1^6	$Grate_1^6=0.63628049$		P_2^6	---

Table 5The first new population $P_{1'}$ in the example

Population	Individuals	Fitness
$P_{1'}$	$P_{1'}^1=(18-16-15-3-2, 18-16-17, 18-19-13-14, 18-19-13-8-9, 18-19-13-12-11), 7-8-9-10-11)$	$f_{1'}^1=0.020053$
	$P_{1'}^2=(11-12-13-3-2, 11-12-13-3-15-16-17, 11-12-13-14, 11-10-9, 11-11)$	$f_{1'}^2=0.020682$
	$P_{1'}^3=(3-2, 3-15-16-17, 3-13-14, 3-13-8-9, 3-13-12-11)$	$f_{1'}^3=0.021774$
	$P_{1'}^4=(8-13-3-2, 8-13-19-18-16-17, 8-13-14, 8-9, 8-9-10-11)$	$f_{1'}^4=0.022096$
	$P_{1'}^5=(13-3-2, 13-19-18-16-17, 13-14, 13-8-9, 13-8-9-10-11)$	$f_{1'}^5=0.022424$
	$P_{1'}^6=(2-2, 2-3-15-16-17, 2-3-13-14, 2-3-13-8-9, 2-3-13-12-11)$	$f_{1'}^6=0.018231$

Table 6The second new population $P_{2'}$ in the example

Population	Individuals	Fitness
$P_{2'}$	$P_{2'}^1=(15-3-2, 15-16-17, 15-3-13-14, 15-3-13-8-9, 15-3-13-12-11)$	$f_{2'}^1=0.01966$
	$P_{2'}^2=(8-7-6-5-3-2, 8-13-3-15-16-17, 8-13-14, 8-9, 8-9-10-11)$	$f_{2'}^2=0.020272$
	$P_{2'}^3=(14-13-3-2, 14-13-3-15-16-17, 14-14, 14-13-8-9, 14-13-12-11)$	$f_{2'}^3=0.021095$
	$P_{2'}^4=(12-13-3-2, 12-13-3-15-16-17, 12-13-14, 12-11-10-9, 12-11)$	$f_{2'}^4=0.022058$
	$P_{2'}^5=(19-13-3-2, 19-18-16-17, 19-13-14, 19-13-8-9, 19-13-12-11)$	$f_{2'}^5=0.022214$
	$P_{2'}^6=(7-6-5-3-2, 7-6-5-3-15-16-17, 7-8-13-14, 7-8-9, 7-8-9-10-11)$	$f_{2'}^6=0.01895$

We observe that except for P_1^6 , the fitness of all individuals in the two populations are improved and $P_{1'}^5$ is the best individual in $P_{1'}$. It is only $P_{1'}^6$ that need carry out the neighborhood learning and the corresponding learning rate can be calculated by Eq. (16), that is, $Nrate_{1'}^6=0.875$.

Then we obtain a new individual $P_{1''}^6=(2-2, 2-3-13-19-18-16-17, 2-3-13-14, 2-3-13-8-9-10, 2-3-13-12-11-10)$, $f_{1''}^6=0.017457$ and $\overline{f_{1''}^6}=0.02074767$. Its fitness is smaller than $f_{1'}^6$, thus it needs to carry out the self-learning, and the corresponding learning rate can be

calculated by Eq. (17), that is, $Srate_1^6 = 0.95080117$.

Then we obtain a new individual $P_1^6 = (12-13-3-2, 12-13-19-18-16-17, 12-13-14, 12-13-8-9, 12-11)$ and $f_1^6 = 0.022634$, $\bar{f}_1^6 = 0.0216105$, $\bar{f}_2^6 = 0.02070817$, $\bar{f}_E^6 = 0.02115934$, $P_1^{best} = P_1^6$, $P_2^{best} = P_2^5$, $P_{best}^m = P_1^6$.

Such iteration is carried out for ten times. We get $P^{best^{10}} = (3-2, 3-13-19-18-16-17, 3-13-14, 3-5-6-7-8-9, 3-5-6-7-8-9-10-11)$ and its rendezvous point is v_{BJ} . Because $P^{best^{10}}$ is loop-free and has no redundant branches, we obtain M^2ST_{RPCS} directly. Further, its QoS = (3Mbps, 140ms, 87ms, 0) satisfies capacity and QoS constraints. M^2ST_{RPCS} is denoted with the yellow lines in Fig. 5.

In the following, we pay attention to the generating process of M^2ST_{OSIS} . The food sources (namely the neighbor nodes) continuously join in M^2ST starting with the rendezvous point v_{BJ} obtained from M^2ST_{RPCS} , shown from Fig. 5 (a)–(f). In more detail, from Fig. 5 (a), the rendezvous point v_{BJ} selects the first three neighbor nodes (v_{SY} , v_{ZZ} , and v_{WH}) with larger SP

($SP_{v_{SY}} = 0.268358$, $SP_{v_{ZZ}} = 0.290527$, $SP_{v_{WH}} = 0.221703$, $SP_{v_{TJ}} = 0.219412$) as the next hop nodes. The number of the selected neighbor nodes depends on that in M^2ST_{RPCS} only at this iteration and depends on Eq. (20) at other iterations; from Fig. 5 (b), v_{SY} selects node v_{DL} ($A_{v_{SY}} = 1$, $SP_{v_{DL}} = 0.62483$, $SP_{v_{CC}} = 0.37517$) as the next hop node, v_{ZZ} selects node v_{XA} ($A_{v_{ZZ}} = 1$, $|NEI_{v_{ZZ}}| = 1$) as the next hop node, v_{WH} selects v_{CS} and v_{NJ} ($A_{v_{WH}} = 2$, $SP_{v_{NJ}} = 0.257143$, $SP_{v_{CS}} = 0.257119$, $SP_{v_{GZ}} = 0.253514$, $SP_{v_{CQ}} = 0.232224$) as the next hop nodes; this process is repeated in Fig. 5 (c)–(e); after all the multicast members are found, the redundant nodes and links, such as v_{SY-DL} and v_{DL} , are removed, and we get $M^2ST_{OSIS} = (3-2, 3-15-16-17, 3-13-14, 3-13-8-9, 3-13-8-9-10-11)$ as shown in Fig. 5 (f), and its fitness is 0.025302, which is larger than that (0.02272) of M^2ST_{RPCS} . Thus, we select M^2ST_{OSIS} as M^2ST_{OPT} .

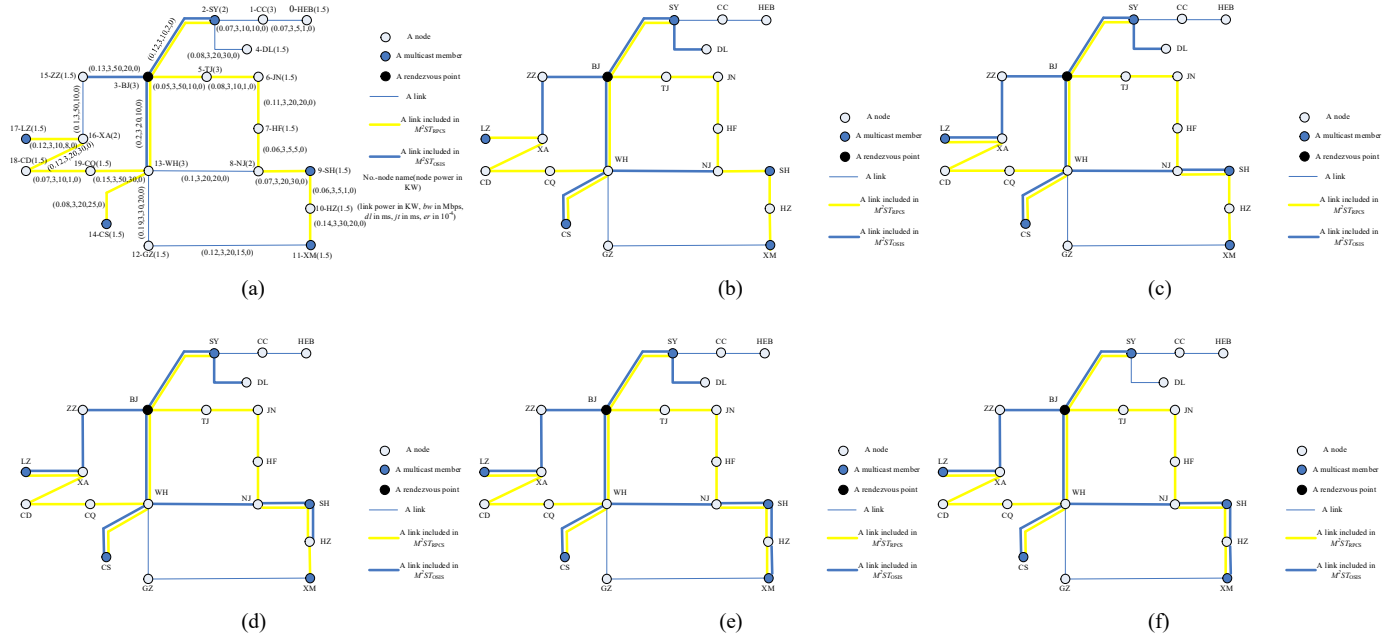


Fig. 5 A simple example

4.3. Time complexity analysis

GIQM initializes the relevant parameters of the algorithm in $O(N_1 + N_2 + \dots + N_k + \dots + N_n) = O(|V|)$ and P_k^i by Dijkstra algorithm in $O(|V|^2)$. In RPCS, at each iteration, it takes $O(N_1 + N_2 + \dots + N_k + \dots + N_n) + O(n)$ to calculate the fitness: f_k^i , \bar{f}_k^i and \bar{f}_E^i (since $n \leq |V|$, $O(|V|)$ is taken);

updating P_k^{best} and P_{best} needs $O(|V|)$ time; for each P_k , P_k^i has the worst case (that is, global learning, neighborhood learning and self-learning are all executed) time complexity of $O(N_k^2)$, and thus for all the populations the worst case time complexity is $O(N_1^2 + N_2^2 + \dots + N_k^2 + \dots + N_n^2)$; calculating new \bar{f}_E and P_{best} requires $O(N_1 + N_2 + \dots + N_k + \dots + N_n) = O(|V|)$. Since the

iterations are repeated I_{\max} times, the total time complexity of RPCS is

$$O\left(I_{\max} \times \left(|V| + |V| + \left(N_1^2 + N_2^2 + \dots + N_k^2 + \dots + N_n^2\right) + |V|\right)\right),$$

that is, $O\left(I_{\max} \times \left(|V| + \left(N_1^2 + N_2^2 + \dots + N_k^2 + \dots + N_n^2\right)\right)\right).$

In OSIS, the worst case time complexity in calculating the selection probability SP_{v_i} of each neighbor node v_i of the current node v_{RP} is $O(|V|)$, and the worst case time complexity in selecting v_i with high probability as the next hop node is $O(|V|^2)$. Therefore, when M^2ST_{OSIS} is generated, the worst case time complexity is $O(|V|(|V| + |V|^2)) = O(|V|^3)$. Thus, in the worst case, GIQM has the time complexity of $O(|V| + |V|^2 + I_{\max} \times (|V| + (N_1^2 + N_2^2 + \dots + N_k^2 + \dots + N_n^2)) + |V|^3)$, that is, $O(I_{\max} \times |V|^3)$.

5. Simulation and evaluation

In this section, simulation configuration is first described in Section 5.1. Then, in Section 5.2, the performance of GIQM is evaluated over realistic topologies with $MLU=0.7$, which is introduced to reflect the influence of network margin on network power consumption and performance. We select the Core-Based Trees (CBT) multicast routing algorithm [49], the Single Shortest Path First (SSPF) point-to-point (P2P) routing algorithm [50] and the constrained Multicast-Max-Flow-Min-Cost algorithm (MMFMC) [45] as the benchmarks due to their similar concerns over multicast routing to our concerns.

CBT is a typical multicast routing algorithm. A CBT has a single node as the core of the tree, from which branches emanate. These branches are made up of other non-core nodes, which form a shortest path between a member of multicast group and the core. The formation of CBT is receiver-based, i.e., a node does not become part of a tree for a particular group unless the node is intended to become a member of that group (or is on the path between a potential member and the tree, in which case the node must become part of the tree). This is of significant benefit to all nodes on the shortest-path between a non-receiver sender and the multicast tree, since they do not incur tree-building overhead. However, the core placement of CBT is usually determined by an external agreement based on the judgement of which node is "known" about the network topology between the current members, even the "best guess" of the core placement are adopted [49]. Therefore, the selection of the core of a CBT is relatively coarse, and its performance is worse than other schemes in many cases.

SSPF is a recently proposed efficient heuristic for energy-aware routing, and is used to solve an ILP optimization

problem, which aims to shut down a subset of bundled links during off-peak periods to minimize energy expenditure. It has three versions: SSPF-1, SSPF-2 and SSPF-R. It uses the greedy heuristic approach to produce a set of paths to route all traffic demands through all powered-on cables in the link set. It firstly uses each shortest path to route the traffic flow of each demand; then, it calculates the total number of cables for each link needed to route all demands in order to switch off the maximal number of unused cables from each link at the same time ensure that the remaining cables are capable of meeting all traffic demands; next, it iteratively selects a candidate link and aims to switch off one of its cables by a function used in SSPF-1 to select the link with the largest spare capacity, or by a function used in SSPF-2 to select the link with the smallest average flow per demand, and for each selected link, a greedy heuristic is used to check if deleting a cable is feasible; finally, it carries out a sequential process of restoring the deleted cables in SSPF-R, repeatedly assuming that a deleted cable leads to a local minimum and the remaining deleted cables are correct decisions to correct the possible mistakes which lead to the local minima.

MMFMC is a one-to-many multicast routing algorithm to achieve the maximum aggregate throughput using network coding while simultaneously achieving the minimum cost, subject to cost constraints. It is based on the constrained Unicast-Max-Flow-Min-Cost algorithm (UMFMC) which maximize the aggregate flow in a graph while minimizing the routing cost. UMFMC accepts constraint on the maximum allowable cost of any unicast traffic flow as its input. For traffic to be delivered, a subgraph containing a set of candidate edges is specified. The removal of undesirable edges results in the specification of a sub-graph for traffic. UMFMC consists of two Linear Programs (LPs). The first LP called the Constrained-Maximum-Flow LP (CMF-LP) maximizes the aggregated traffic flow, subject to the constraint that traffic is routed over its subgraph. The allowable cost for every traffic flow is constrained by selecting the subgraph appropriately. The second LP called the Constrained-Minimum-Cost LP (CMC-LP) minimizes the cost of the maximum aggregated traffic flow, subject to the constraint that traffic is routed over its subgraph.

To evaluate the performance of GIQM, we investigate the six devised routing schemes (denoted as Scheme 1-6 in the following) and introduce power consumption into their routing process to carry out the reasonable comparison with our proposed scheme (denoted as Scheme 7). All these schemes are described briefly as follows.

Scheme 1: SSPF-M. Since SSPF was devised for P2P communication, we have to extend it to support the many-to-many multicast scenario and denote it as SSPF-M. At first, a one-to-many multicast tree is made up of a set of shortest paths produced by SSPF-2 between each multicast sender and the multicast receivers. Then all the one-to-many multicast trees are merged and redundant branches are cut to construct a many-to-many multicast tree. Finally, SSPF-R is used to repeatedly check the feasibility of deleting a cable and avoid local minima.

Scheme 2: CBT. Referring to the core placement methods in [49], the node with the minimum average hops to all the multicast group members is selected as the core of the multicast group. The many-to-many multicast tree formed is a shortest-path spanning tree (SPT) rooted at the core.

Scheme 3: C-S. We incorporate the main idea of SSPF into CBT. The core placement is the same as that in Scheme 2. The many-to-many multicast tree is made up of a set of shortest paths produced by SSPF-2 between the core and the multicast members. SSPF-R is used to avoid the local minima.

Scheme 4: G-S. We take the rendezvous point obtained in RPCS as the root of the multicast tree, and the generating process of the many-to-many multicast tree is the same as that in the Scheme 3.

Scheme 5: G-C. Taking the rendezvous point obtained in RPCS as the core, the core-based multicast tree is generated just as that in Scheme 2.

Scheme 6: MMFMP. We set power consumption of each link respectively as the corresponding edge cost, and then MMFMC can be further denoted as MMFMP. MMFMP is developed to solve the one-to-many multicast routing problem, so we have to simply merge the multicast trees obtained by MMFMP as the

solution under many-to-many multicast routing scenario after cutting the redundant branches.

Scheme 7: our proposed GIQM.

5.1. Simulation configuration

5.1.1. Development Environment

The configurations of hardware and software used for simulation are as follows:

CPU: Intel Quad-Core i5-4590 @ 3.30GHz;

RAM: 4GB (DDR3, 1600MHz);

OS: Windows 8.1 professional 64bit;

IDE: Microsoft Visual Studio 2010.

5.1.2. The topology of the simulation platform

In this paper, the performance of GIQM is evaluated over the topologies of Cernet2 (China Education and Research Network 2) [51], GéANT [52] and Internet2 [53]. The topologies of the simulation platform and the comparison on characteristics of topologies under consideration are shown in Fig. 6 and Table 7, respectively.

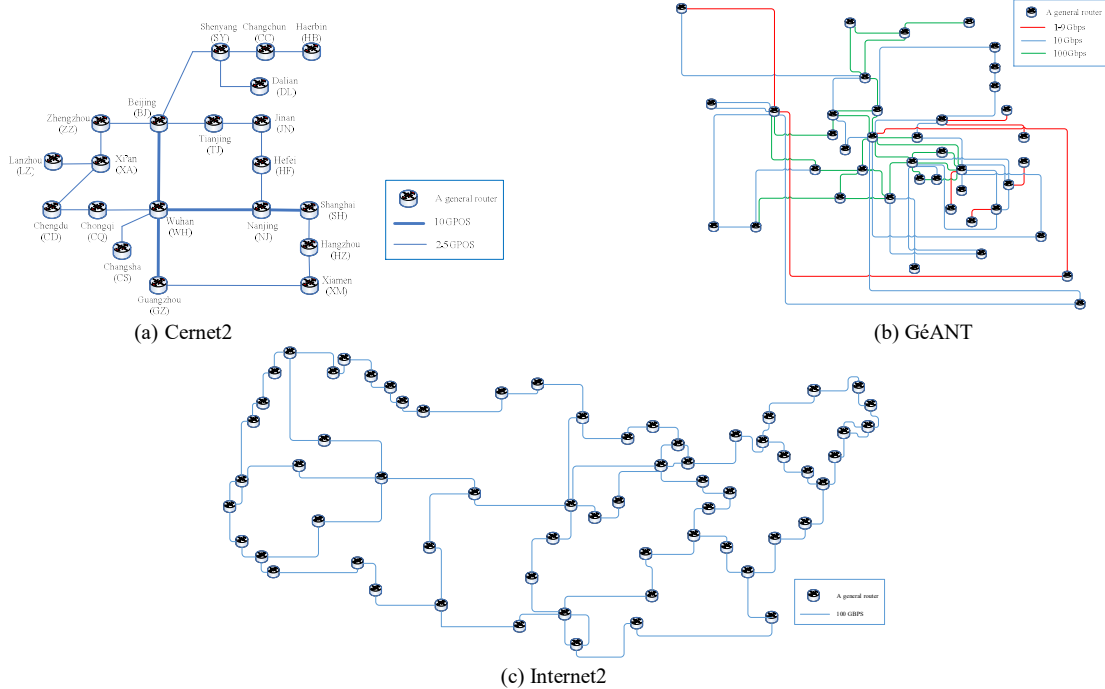


Fig. 6 Network platform

Table 7
Comparison on characteristics of topologies under consideration

Characteristics	Cernet2	GéANT	Internet2
The total number of nodes	20	41	64
The total number of links	22	65	78
The number of links with capacity less than 10Gbps	4	8	0
The number of links with capacity between 10Gbps and 100Gbps	18	30	0
The number of links with capacity more than 100Gbps	0	27	78

5.1.3. Scenario configuration

In all simulation scenarios in this paper, the nodes are assumed to have the same energy profile. Every node represents a core router, e.g., the Cisco 12000 core router [54]. It is assumed that 4 chassis are included in a router, 4 line cards are included in a chassis and 4 ports are included in a line card in the network. The constants α and β in the power model of a node are respectively given the value 0.032 and 0.82 according to Vereecken et al. [55]. By referring to Cisco XR 12000 Series and Cisco 12000 Series Routers [54], the settings of power consumption parameters used in the simulation are shown in Table 8. Our comparison and evaluation among the above seven schemes are carried out with $MLU=0.7$, as shown in Section 5.2.

Table 8
Power consumption parameter values used in simulation

Parameter	Value
Power consumption of a master engine	356W
Power consumption of a chassis	100W
Power consumption of a forwarding engine	446W
Power consumption of a replication engine	100W
Power consumption of a port	150W
Power consumption of an optical pre-amplifier	4.8W
Power consumption of an optical in-line amplifier	10W
Power consumption of an optical post-amplifier	4.8W
Power consumption of an optical regenerator	26W

For the traffic matrix used in simulation, referring to the approach from [56], it has been obtained from the "Nobel-2 undirected graph" data file which is publicly available in [57] containing the undirected traffic demand between each pair of nodes. The directed traffic demands can be obtained by randomly split the demand between two directions. Referring to that in the *nobel-eu* topology [57] (including 28 nodes and 41 links) shown in Fig. 7, the total amount of traffic demands in Cernet2, Internet2 and GÉANT are respectively set to 55 Gb/s, 5Tb/s and 2Tb/s, distributed among the origin-destination pairs.

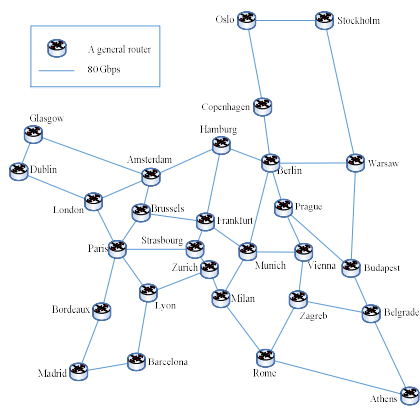


Fig. 7 Nobel-eu topology

To identify the relative importance of different QoS parameters to the user, we refer to the standards recommended in ITU-T Y.1541 [48] and the weight for QoS parameters used

in [58], and set $\omega_{bw}=0.111$ for bandwidth, $\omega_{dl}=0.149$ for delay, $\omega_{jt}=0.151$ for delay jitter and $\omega_{er}=0.589$ for error rate.

5.2. Comparison and evaluation

We evaluate Schemes 1–7 on network power consumption, routing success rate, user satisfaction degree on QoS and running time under $MLU=0.7$ over topologies under consideration.

The comparison results on network power consumption from Fig. 8 (a)–(c) indicate that the network power consumption gradually increases along with the rising of the number of multicast members. Furthermore, we also observe that the impact of network scale and network complexity on the power saving potential of Schemes 1–7 from Fig. 8, that is, each scheme over a complex topology shows its better power saving potential than that over a simple one, owing to more redundant components being put into the sleeping state over the former than those over the latter.

For all of the three network topologies, the power consumption of GIQM is the lowest among all the schemes while the power consumption of SSPF is the highest. The gaps among the schemes over the same topology mainly depend on different routing strategies. The power saving in SSPF relies on shutting down a subset of the bundled links as many as possible by the Greedy Heuristics. Although we improve SSPF to support the multicast scenario, its underlying unicast routing property makes the multicast data packets travel on some extra paths, which leads to unnecessary power consumption. In CBT, a core-based forwarding tree is established to meet the multicast communication. One major advantage is that the adoption of the core-based idea for the multicast communication helps in optimizing the routes between any sender and group members on the tree, and thus facilitates power saving under the multicast scenario. Another major advantage of CBT is that the CBT formation and multicast packet flow are decoupled from, but take full advantage of, underlying unicast routing, irrespective of which underlying unicast algorithm is operating. These factors result in the CBT being more power-saving than SSPF. For GIQM, the optimized many-to-many multicast tree is obtained by comparing solutions from two stages, that is, the candidate multicast tree derived at the first RPCS stage and the modified multicast tree rooted at the exquisite rendezvous points derived at the second OSIS stage, which can avoid the local optimal solution as much as possible. Such a two-stage solving process enables GIQM to have the most prominent power-saving potential in all the above schemes. The C-S, G-S and G-C schemes are hybrid schemes on the basis of SSPF, CBT and GIQM. C-S and G-S are only different in the selection of the core or the rendezvous point. When the core selected in C-S and the rendezvous point selected in G-S are the same node, the power consumption of C-S and G-S is equal, e.g., the number of multicast members is 4 and 12 over Cernet2 in Fig. 8 (a); otherwise, in most cases, the power consumption of G-S is lower than that of C-S except that the number of multicast members is 10 and 12 over GÉANT in Fig. 8 (b), owing to the fact that using the rendezvous point obtained in RPCS can always get a better core than that of C-S.

However, neither G-S nor C-S is more power-saving than CBT. Similarly, when the core selected in CBT and the rendezvous point selected in G-C are the same node, the power consumption of G-C and CBT is equal; otherwise, the power consumption of G-C is lower than that of CBT. The power consumption of MMFMP is between that of G-S and CBT, and

the result is mainly because that MMFMP achieves partial traffic aggregation using network coding in the process of generating one-to-many multicast trees and simple merging of one-to-many multicast trees in the process of generating the many-to-many multicast solution, which make it much better than G-S and much worse than CBT.

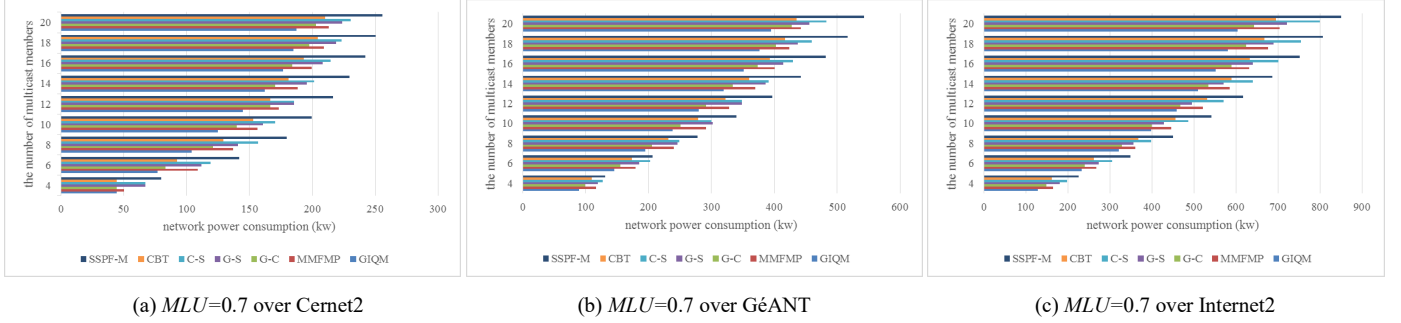


Fig. 8 Performance comparison on network power consumption

The comparison results on routing success rate from Fig. 9 (a)–(c) indicate that along with the increasing number of multicast members, network resources become deficient when searching paths that satisfy QoS demands for all services and hence routing success rate is gradually decreasing in each scheme. Furthermore, we also observe the impact of network scale and network complexity on the routing success rate of Schemes 1–7 from Fig. 9, that is, each scheme over Cernet2 attains the greatest routing success rate over all the three topologies and each one in Internet2 is the worst, mainly owing to deterioration of QoS guarantee in the complex topology.

Furthermore, for all of the three network topologies the routing success rate of GIQM is higher and also its decrease rate with the increase of the number of multicast members is slower than those of the other schemes, which is due to the user satisfaction degree on QoS being considered in the routing process and QoS being supported flexibly in the interval form to enable adjusting route adaptively. In CBT, the worst situation in the routing success rate occurs when the traffic demands are raising sharply, the multicast tree, especially its core, exhausts itself to cope with the huge traffic due to its coarse core selection and its lacking of QoS guarantee. As for SSPF, although QoS consideration is still missing, the underlying

unicast routing property of SSPF makes it, to some extent, be scalable to the increasing traffic demands and thus avoid the traffic being concentrated on the single node or path at the cost of much power consumption. In other schemes, we can see that the G-S scheme is better than C-S, G-C and SSPF due to its partial combination of QoS guarantee and underlying unicast routing property. Furthermore, G-C is better than SSPF and C-S, because G-C takes QoS guarantee into account in the process of determining the rendezvous point in RPCS. For the comparison between SSPF and C-S, when the number of multicast members is more than 8, SSPF shows a significantly better routing success rate than C-S, because the core of the multicast tree in C-S is becoming more and more exhausted with the increasing traffic flows and even gets congested; in other cases, there is little difference between the two schemes. As for MMFMP, we can observe that its routing success rate is between that of CBT and C-S, which is mainly because that the simple merging of one-to-many multicast trees in MMFMP makes more incoming traffic accommodated in the many-to-many multicast solution compared with CBT, and the aggregation of unicast flows achieved by using network coding leads to less remaining capacity and worse QoS guarantee than that of C-S.

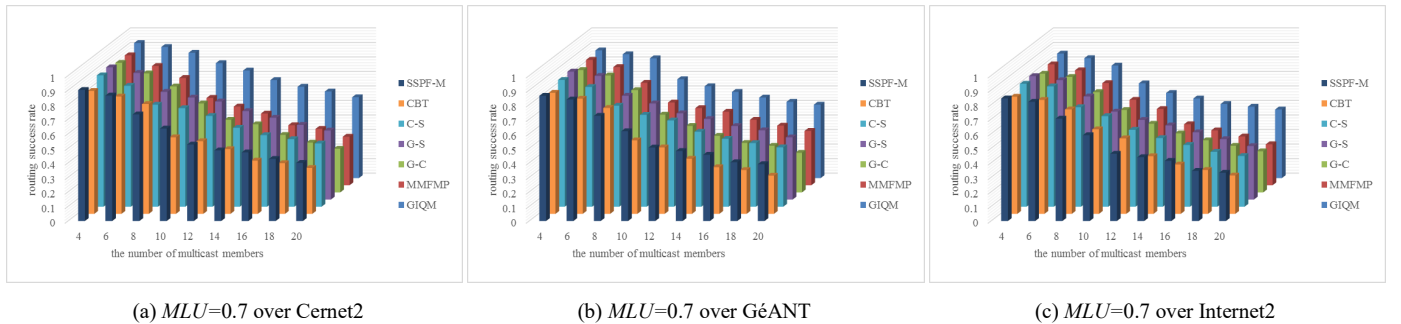


Fig. 9 Performance comparison on routing success rate

The comparison results on the user satisfaction degree on QoS from Fig. 10 (a)–(c) indicate that the user satisfaction

degree on QoS gradually decreases with the increase of the number of multicast members. Furthermore, we also observe the impact of network scale and network complexity on the user satisfaction degree on QoS for Schemes 1–7 in Fig. 10, that is, QoS for each scheme over Cernet2 are the greatest in all the three topologies and that over Internet2 are the worst, mainly owing to significant increases in delay, delay jitter and error rate for the optimal M^2ST over a complex topology.

Furthermore, for all of the three network topologies, the user satisfaction degree on QoS in GIQM is higher and also their decrease rates are slower than those of the other schemes, because in GIQM the QoS satisfaction degree is used in routing decision and the tree with the best user satisfaction degree on QoS is always chosen as the route for the incoming many-to-many traffic demand. In Fig. 10, it can be observed that the user satisfaction degree on QoS of CBT is the worst among all the schemes. The reason is that the generating process of its multicast tree does not support QoS guarantee and the coarse selection of the core makes it difficult to process relatively large traffic demands timely. For SSPF, it does not consider QoS when routing, however, its user satisfaction

degree on QoS is far better than that of CBT, mainly thanking to its underlying unicast routing property, which enables it to avoid the traffic concentrated on a single node or path and thus improves QoS to some extent. In other schemes, we can see that the G-S scheme is better than C-S and G-C owing to its partial combination of the exquisite core choice in RPCS and underlying unicast routing property from SSPF. Furthermore, G-C is better than SSPF and C-S because of its exquisite core (rendezvous point in RPCS) selection. For the gap of user satisfaction degree on QoS between SSPF and C-S, it is mainly because the underlying unicast routing property of SSPF is superior to the negative effect brought by the coarse selection of the core of C-S. As for MMFMP, we can observe that its user satisfaction degree on QoS is between that of CBT and C-S. The reason is that network coding used in MMFMP can usually achieve high throughput (leading to the high user satisfaction degree on bandwidth) and low error rate (leading to the high user satisfaction degree on error rate), but simultaneously cause extra delay due to encoding and decoding operations (leading to the low user satisfaction degree on delay and jitter).

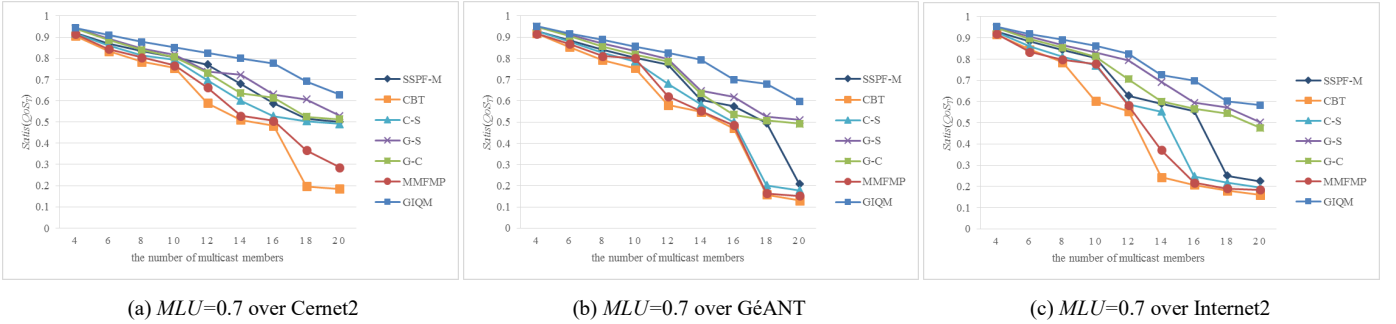


Fig. 10 QoS satisfaction degree comparison

The comparison results on the running time from Fig. 11 (a)–(c) indicate that along with the increasing number of multicast members, the running time gradually increases in each scheme. Furthermore, we also observe the impact of network scale and network complexity on the running time of Schemes 1–7 from Fig. 11, that is, each scheme over Cernet2 takes the least time in all the three topologies and each one over Internet2 takes the most time, mainly owing to the expansion of the solution space and the increase of computation expenditure in the complex topology.

As expected, the running time of GIQM is higher than that of the other schemes, due to the time-consuming evolution stage of RPCS and the time-consuming bio-inspired meta-heuristic stage of OSIS. While in SSPF, in order to shut down a subnet of the bundled links, SSPF-2 uses the greedy heuristic function to check if deleting a fiber in a link is feasible and then SSPF-R heuristically restores all fibers in a link and select a candidate

fiber in another link to avoid local minima. Hence, the running time of SSPF is higher than the remaining other five schemes although it is lower than that of GIQM. For CBT, its running time is lower than the hybrid schemes (C-S, G-S and G-C) and even far less than that of the schemes as a result of not using any time-consuming solving methods in the routing process. In the hybrid schemes, arising from the rendezvous point selection as that in GIQM and then the multicast tree generated as that in SSPF, G-S has a higher running time than C-S and G-C. Furthermore, from Fig. 11, we observe that G-C has a longer running time than SSPF and the gap will become even larger when the number of multicast members increases. For C-S scheme, its running time is between that of SSPF and CBT. As for MMFMP, its running time is the lowest in all the schemes, even much less than that of CBT, owing to its polynomial-time solutions.

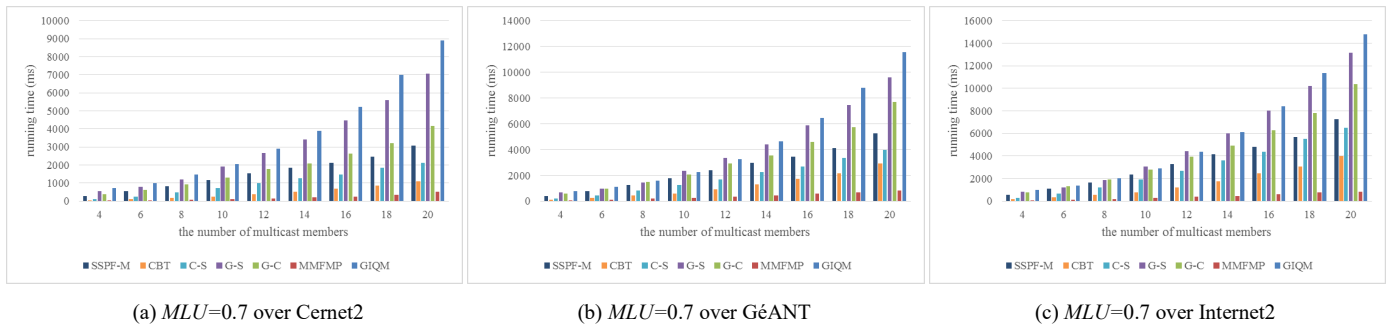


Fig. 11 Performance comparison on the running time

6. Conclusions

This paper focuses on power saving and QoS satisfaction for many-to-many multicast scenario in the backbone network. We devise a two-stage power-efficient QoS algorithm, that is, GIQM, to solve the many-to-many multicast routing problem, which is formulated as a nonlinear programming problem mathematically. In particular, we elaborate power profiles of network elements explicitly and enable power consumption as a metric for routing. Also, the user QoS requirements are flexibly supported in the form of interval, and a user satisfaction degree function on QoS is defined based on a fuzzy membership degree method. The proposed GIQM is compared with the other six routing schemes on power consumption, routing success rate and running time. At the cost of running time, GIQM achieves the best performance among all the schemes, indicating that it is feasible and effective in terms of power-saving potential and routing quality.

Furthermore, a more in-depth subsequent study will be carried out to pursue more significant power saving in the future. The power profiles of network elements will be characterized more elaborately to enable much more effective and efficient routing decision. Moreover, a more power-saving QoS many-to-many multicast routing algorithm will also be exploited to solve the proposed problem in this paper much more efficiently, especially improving the running time of the proposed solution.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 61572123; the National Science Foundation for Distinguished Young Scholars of China under Grant No. 71325002.

The authors would like to thank the anonymous reviewers for their comments to improve the paper.

References

- [1] Joseph Chabarek, Joel Sommers, Paul Barford, Cristian Estan, David Tsang, Stephen Wright. Power awareness in network design and routing [C]. The 27th Conference on Computer Communications, IEEE INFOCOM 2008, Pages: 1130-1138.
- [2] Rafiullah Khan, Raffaele Bolla, et al. Smart proxying for reducing network energy consumption [C]. 2012 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS), Pages: 1-8.
- [3] W.Vereecken, L.Deboosere, D.Colle, et al. Energy efficiency in telecommunication networks [C]. 13th European Conference on Networks and Optical Communications (NOC), 2008, Pages: 44-51.
- [4] Ward Van Heddeghem, Sofie Lambert, Bart Lannoo, Didier Colle, Mario Pickavet, Piet Demeester. Trends in worldwide ICT electricity consumption from 2007 to 2012 [J]. Computer Communications, 2014, 50: 64-76.
- [5] T.A. Napp, A. Gambhir, T.P. Hills, N. Florin, P.S Fennell. A review of the technologies, economics and policy instruments for decarbonising energy-intensive manufacturing industries [J]. Renewable and Sustainable Energy Reviews, 2014, 30: 616-640.
- [6] Murat Kocaoglu, Derya Malak, Ozgur B. Akan. Fundamentals of Green Communications and Computing: Modeling and Simulation [J]. Computer, 2012, 45(9): 40-46.
- [7] Global e-Sustainability Initiative (GeSI), SMART 2020: Enabling the Low Carbon Economy in the Information Age [EB/OL]. <<http://www.theclimategroup.org/assets/resources/publications/Smart2020Report.pdf>>.
- [8] Maruti Gupta, Suresh Singh. Greening of the internet [C]. SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, Pages: 19-26.
- [9] Will Fisher, Martin Suchara, Jennifer Rexford. Greening backbone networks: reducing energy consumption by shutting off cables in bundled links [J]. Green Networking '10: Proceedings of the first ACM SIGCOMM workshop on Green networking, 2010, Pages: 29-34.
- [10] Bianzino, A.P., Chaudet, C., Rossi, D., Rougier, J.L.. A Survey of Green Networking Research [J]. IEEE Communications Surveys & Tutorials, 2012, 14(1): 3-20.
- [11] Christoph Lange, Dirk Kosiankowski, Rainer Weidmann, Andreas Gladisch. Energy Consumption of Telecommunication Networks and Related Improvement Options [J]. IEEE Journal of Selected Topics in Quantum Electronics, 2011, 17(2): 285-295.
- [12] Raffaele Bolla, Roberto Bruschi, Franco Davoli, Flavio Cucchietti. Energy Efficiency in the Future Internet: A Survey of Existing Approaches and Trends in Energy-Aware Fixed Network Infrastructures [J]. IEEE Communications Surveys & Tutorials, 2011, 13(2): 223-244.
- [13] Sherif A. Tawfik, Volkan Kursun. Dual Supply Voltages and Dual Clock Frequencies for Lower Clock Power and Suppressed Temperature-Gradient-Induced Clock Skew [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2010, 18(3): 347-355.
- [14] Quansheng Guan, Tianyu Chen, Shengming Jiang, Fei Ji, Fangjiong Chen. User-Network Cooperation-Based Sleep Scheduling for Communication Networks [J]. IEEE Journal on Selected Areas in Communications, 2016, 34(12): 3313-3325.
- [15] Xuejiao Zhao, Gangxiang Shen, Weidong Shao, Limei Peng. Energy-Minimized Design and Operation of IP Over WDM Networks With Traffic-Aware Adaptive Router Card Clock Frequency [J]. IEEE Journal on Selected Areas in Communications, 2015, 33(12): 2847-2862.
- [16] Antonio Cianfrani, Vincenzo Eramo, Marco Listanti, Marco Poverini. An OSPF enhancement for energy saving in IP networks [C]. 2011 IEEE

- Conference on Computer Communications Workshops (INFOCOM WKSHPS), 2011, Pages: 325-330.
- [17] Xingwei Wang, Dapeng Qu, Min Huang, Keqin Li, Sajal K. Das, Jinhong Zhang, Ruiyun Yu. Multiple many-to-many multicast routing scheme in green multi-granularity transport networks [J]. *Computer Networks*, 2015, 93(P1): 225-242.
 - [18] E. Amaldi, A. Capone, L.G. Gianoli. Energy-aware IP traffic engineering with shortest path routing [J]. *Computer Networks*, 2013, 57(6): 1503-1517.
 - [19] Chengchen Hu, Chunming Wu, Wei Xiong, Binqiang Wang, Jiangxing Wu, Ming Jiang. On the design of green reconfigurable router toward energy efficient internet [J]. *IEEE Communications Magazine*, 2011, 49(6): 83-87.
 - [20] Melody Moh, Bang Nguyen. QoS-guaranteed one-to-many and many-to-many multicast routing [J]. *Computer Communications*, 2003, 26(7): 652-669.
 - [21] Adel Ben Mnaouer, Abdelfettah Belghith, Mohamed Aissa. New Strategies and Approaches for Efficient Overlay Multicast Routing [J]. *Procedia Computer Science*, 2012, 10: 161-169.
 - [22] A.T. Haghighat, K. Faez, M. Dehghan, A. Mowlaei, Y. Ghahremani. GA-based heuristic algorithms for bandwidth-delay-constrained least-cost multicast routing [J]. *Computer Communications*, 2004, 27(1): 111-127.
 - [23] Jiyun Wu, Yanlei Shang, Chau Yuen, Bo Cheng, Junliang Chen. TRADER: A reliable transmission scheme to video conferencing applications over the internet [J]. *Journal of Network and Computer Applications*, 2014, 44: 161-171.
 - [24] Tim Szegedi, Christina Hattingh, Robert Barton, Kenneth Briley. End-to-end QoS Network Design: Quality of Service for Rich-Media & Cloud Networks [M]. Cisco Press, 2013.
 - [25] Michael Jarschel, Daniel Schlosser, Sven Scheuring, Tobias Hofffeld. Gaming in the clouds: QoE and the users' perspective [J]. *Mathematical and Computer Modelling*, 2013, 57(11-12): 2883-2894.
 - [26] X. Jia, N. Pissinou, K. Makki. A real-time multicast routing algorithm for multimedia applications [J]. *Computer Communications*, 1997, 20(12): 1098-1106.
 - [27] Zhou C H, Xie A S. Dynamic niche-based self-organizing learning algorithm [J]. *Journal of Software*, 2011, 22(8): 1738-1748, (in Chinese).
 - [28] Taherdangkoo M, Shirzadi M H, Bagheri M H. A novel meta-heuristic algorithm for numerical function optimization: Blind, naked mole-rats (BNMR) algorithm [J]. *Scientific Research and Essays*, 2012, 7(41): 3566-3583.
 - [29] Xingwei Wang, Min Huang, Jiren Liu. Research on quality of service based destination node joining and leaving algorithm for multimedia group communication [J]. *Chinese Journal of Computers*, 2001, 24(8): 838-844, (in Chinese).
 - [30] R. Leela, N. Thanulekshmi, S. Selvakumar. Multi-constraint QoS Unicast Routing Using Genetic Algorithm (MURUGA) [J]. *Applied Soft Computing*, 2011, 11(2): 1753-1761.
 - [31] Meie Shen, Zhihui Zhan, Weineng Chen, Yuejiao Gong, Jun Zhang, Yun Li. Bi-Velocity Discrete Particle Swarm Optimization and Its Application to Multicast Routing Problem in Communication Networks [J]. *IEEE Transactions on Industrial Electronics*, 2014, 61(12): 7141-7151.
 - [32] Peng-Yeng Yin, Ray-I. Chang, Chih-Chiang Chao, Yen-Ting Chu. Niche ant colony optimization with colony guides for QoS multicast routing [J]. *Journal of Network and Computer Applications*, 2014, 40: 61-72.
 - [33] Xingwei Wang, Hui Cheng, Min Huang. Multi-robot navigation based QoS routing in self-organizing networks [J]. *Engineering Applications of Artificial Intelligence*, 2013, 26(1): 262-272.
 - [34] Zhixin Sun, Yuhua Xu, Jing Gong, Zhiguang Ren. A multicast routing algorithm based on searching in directed graph [J]. *Applied Mathematics and Computation*, 2011, 218(7): 3723-3732.
 - [35] Hua Wang, Hong Xu, Shanwen Yi, Zhao Shi. A tree-growth based ant colony algorithm for QoS multicast routing problem [J]. *Expert Systems with Applications*, 2011, 38(9): 11787-11795.
 - [36] Moonseong Kim, Hyunseung Choo, Matt W. Mutka, Hyung-Jin Lim, Kwangjin Park. On QoS multicast routing algorithms using k-minimum Steiner trees [J]. *Information Sciences*, 2013, 238: 190-204.
 - [37] Tzu-Lun Huang, D.T. Lee. A distributed multicast routing algorithm for real-time applications in wide area networks [J]. *Journal of Parallel and Distributed Computing*, 2007, 67(5): 516-530.
 - [38] Ahmed Y. Al-Dubai, Mohamed Ould-Khaoua, Lewis M. Mackenzie. Trade-Offs between Latency, Complexity, and Load Balancing with Multicast Algorithms [J]. *IEEE Transactions on Computers*, 2010, 59(2): 159-173.
 - [39] Ahmed Younes. Multicast routing with bandwidth and delay constraints based on genetic algorithms [J]. *Egyptian Informatics Journal*, 2011, 12(2): 107-114.
 - [40] Ta Anh Son, Pham Dinh Tao, Le Thi Hoai An, Khadraoui Djamel. Solving Many to many multicast QoS routing problem using DCA and proximal decomposition technique [C]. *International Conference on Computing, Networking and Communications (ICNC)*, IEEE, 2012, Pages: 809-814.
 - [41] Hung-ying Tyan, Hou Jennifer C., Bin Wang. Many-to-many multicast routing with temporal quality of service guarantees [J]. *IEEE Transactions on Computers*, 2003, 52(6): 826-832.
 - [42] Frédéric Giroire, Joanna Moulrierac, Truong Khoa Phan, Frédéric Roudaut. Minimization of network power consumption with redundancy elimination [J]. *Computer Communications*, 2015, 59: 98-105.
 - [43] Qi Li, Mingwei Xu, Yuan Yang, Lixin Gao, Yong Cui, Jianping Wu. Safe and Practical Energy-Efficient Detour Routing in IP Networks [J]. *IEEE/ACM Transactions on Networking*, 2014, 22(6): 1925-1937.
 - [44] Yuan Yang, Mingwei Xu, Dan Wang, Suogang Li. A Hop-by-Hop Routing Mechanism for Green Internet [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2016, 27(1): 2-16.
 - [45] Ted H. Szymanski. Max-Flow Min-Cost Routing in a Future-Internet with Improved QoS Guarantees [J]. *IEEE Transactions on Communications*, 2013, 61(4): 1485-1497.
 - [46] Xingwei Wang, Hui Cheng, Keqin Li, Jie Li, Jiajia Sun. A cross-layer optimization based integrated routing and grooming algorithm for green multi-granularity transport networks [J]. *Journal of Parallel and Distributed Computing*, 2013, 73(6): 807-822.
 - [47] Xingwei Wang, Jiannong Cao, Hui Cheng, Min Huang. QoS multicast routing for multimedia group communications using intelligent computational methods [J]. *Computer Communications*, 2006, 29(12): 2217-2229.
 - [48] Network Performance Objectives for IP-based Services [S]. ITU-T Y.1541, 2011.
 - [49] Tony Ballardie, Paul Francis, Jon Crowcroft. Core based trees (CBT) [C]. SIGCOMM '93: Conference proceedings on Communications architectures, protocols and applications, 1993, Pages: 85-95.
 - [50] Lin G, Soh S, Chin K W, et al. Efficient Heuristics for Energy-Aware Routing in Networks with Bundled Links [J]. *Computer Networks*, 2013, 57(8): 1774-1788.
 - [51] http://www.edu.cn/xxh/ji_shu_ju_le_bu/cernet2_ipv6/cernet2/
 - [52] http://www.geant.org/Networks/Pan-European_network/Pages/GEANT_topology_map.aspx
 - [53] https://www.internet2.edu/media/medialibrary/2015/08/04/NetworkMap_all.pdf
 - [54] Cisco XR 12000 Series and Cisco 12000 Series Routers [Online] Available: <http://www.cisco.com/c/en/us/products/routers/12000-series-routers/data-sheet-listing.html>.
 - [55] Vereecken W, Van H, Puype B, et al. Optical Networks: How Much Power Do They Consume and How Can We Optimize This [A]. *The 36th European Conference and Exhibition on Optical Communication*, Torino, Italy, 2010, Pages: 1-4.
 - [56] Rosario G. Garroppo, Stefano Giordano, Gianfranco Nencioni, Maria Grazia Scutellà. Mixed Integer Non-Linear Programming models for Green Network Design [J]. *Computers & Operations Research*, 2013, 40(1): 273-281.
 - [57] Simple Network Description Library (SNDlib); 2011. <http://sndlib.zib.de>.
 - [58] Hyun Jong Kim, Dong Geun Yun, Hwa-Suk Kim, Kee Seong Cho, Seong Gon Choi. QoE assessment model for video streaming service using QoS parameters in wired-wireless network [C]. *14th International Conference on Advanced Communication Technology (ICACT)*, 2012, Pages: 459-464.